SECTION 13
EXTERNAL BUS INTERFACE

The MPC823 bus is synchronous and burstable. Signals driven on this bus are required to make the setup and hold time relative to the bus clock's rising edge. This bus has the ability to support multiple masters. The MPC823 architecture supports byte, half-word, and word operands allowing access to 8-,16-, and 32-bit data ports through the use of synchronous cycles controlled by the transfer size output (TSIZx) signals. The slave access to 16- and 8-bit ports is controlled by the memory controller.

13.1 FEATURES

The following is a list of the bus interface's main features:

- 26-Bit Address Bus with Transfer Size Indication
- 32-Bit Data Bus
- TTL-Compatible Interface
- Internal On-Chip Bus Arbitration Logic Supports One External Bus Master
- Chip-Select and Wait State Generation
- Supports Many Different Memory Types
- Asynchronous DRAM Interface Support
- Flash ROM Programming Support
- Compatible with PowerPC Architecture
- Interfaces to Slave Devices Easily
- Synchronous Bus Operation
- Data Parity Support

13.2 TRANSFER SIGNALS

The bus transfers information between the MPC823 and the external memory or peripheral device. External devices can accept or provide 8,16, and 32 bits in parallel and must follow the handshake protocol. The maximum number of bits accepted or provided during a bus transfer is defined as the port width.

The MPC823 contains an address bus that specifies the transfer's address and a data bus that transfers the data. Control signals indicate the beginning and type of the cycle, as well as the address space and size of the transfer. The selected device then controls the length of the cycle with the signal used to terminate the cycle. A strobe signal for the address bus indicates the validity of the address and provides timing information for the data. The MPC823 bus is synchronous, but the bus and control input signals must be timed to setup and hold times relative to the rising edge of the clock. In this situation, bus cycles can be completed in two clock cycles.

Furthermore, for all inputs, the MPC823 latches the level of the input during a sample window around the rising edge of the clock signal. This window is illustrated in Figure 13-1, where **tsu** and **tho** are the input setup and hold times, respectively. To ensure that an input signal is recognized on a specific falling edge of the clock, the input must be stable during the sample window. If an input makes a transition during the window time period, the level recognized by the MPC823 is not predictable. However, the MPC823 always resolves the latched level to either a logic high or low before using it. In addition to meeting input setup and hold times for deterministic operation, all input signals must obey the protocols described in this section.

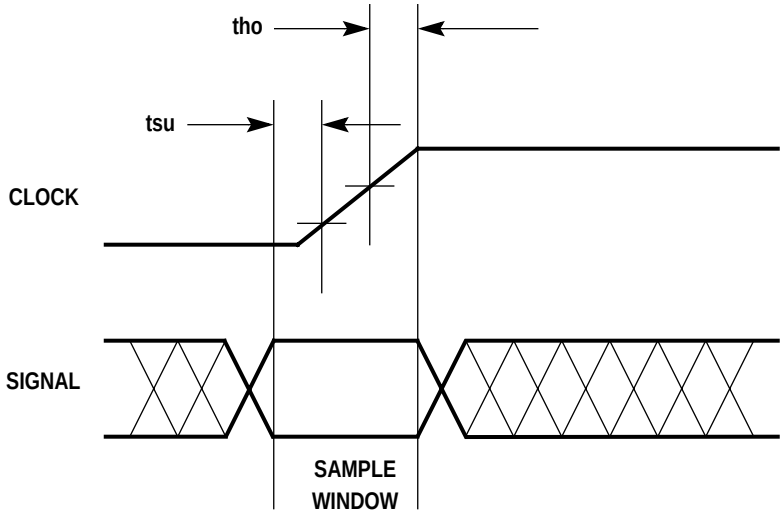

Figure 13-1. Input Sample Window

13.2.1 Control Signals

The MPC823 initiates a bus cycle by driving the address, size, address type, cycle type, and read/write outputs. At the beginning of a bus cycle, the TSIZ0 and TSIZ1 signals are driven with the AT signals. TSIZ0 and TSIZ1 indicate the number of bytes to be transferred during an operand cycle that consists of one or more bus cycles. These signals are valid at the rising edge of the clock in which the $\overline{TS}$ signal is asserted. The RD/$\overline{WR}$ signal determines the direction of the transfer during a bus cycle. Driven at the beginning of a bus cycle, RD/$\overline{WR}$ is valid at the rising edge of the clock in which the $\overline{TS}$ signal is asserted. However, RD/$\overline{WR}$ only transitions when a write cycle is preceded by a read cycle or vice versa. The signal may remain low for consecutive write cycles.

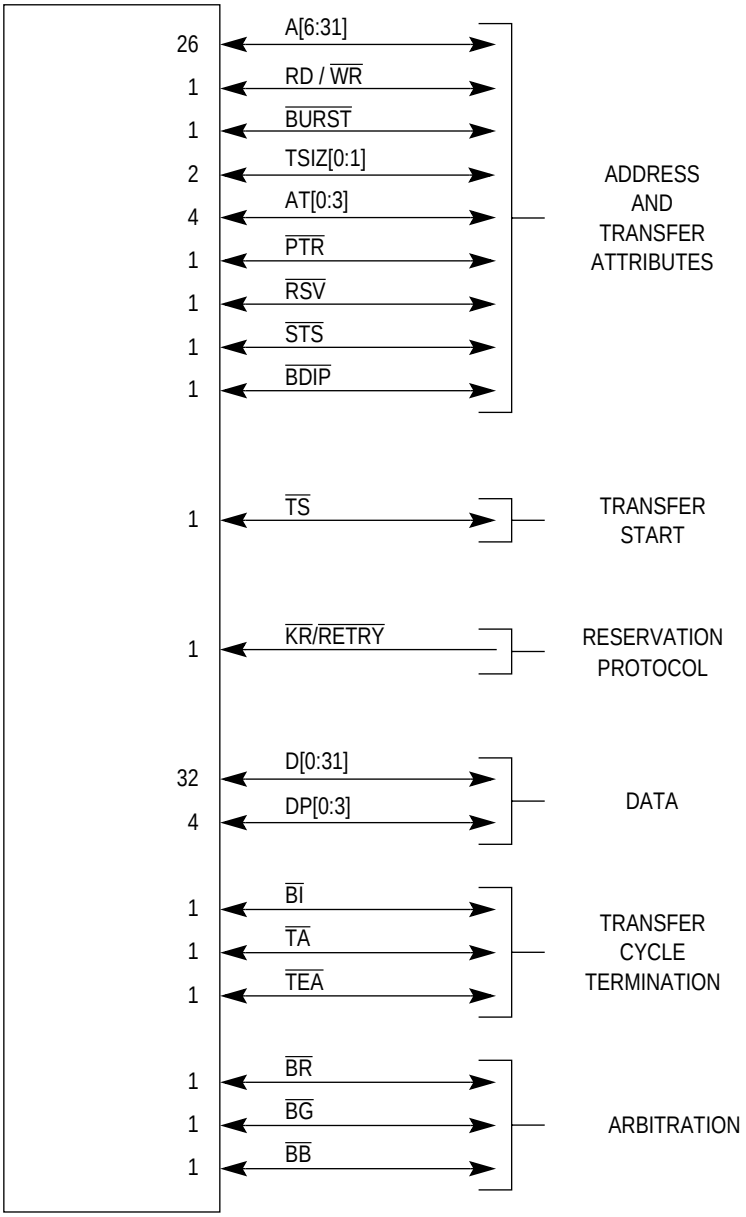

Figure 13-2. MPC823 Bus Signals

13.3 BUS SIGNAL DESCRIPTIONS

The following table decribes each bus interface signal. More detailed descriptions can be found in subsequent sections of this manual.

Table 13-1. Bus Interface Signals

MNEMONIC	PINS	ACTIVE	I/O	DESCRIPTION
ADDRESS AND TRANSFER ATTRIBUTES				
A[6:31]	26	High	O	**Address Bus**—Driven by the MPC823 when it owns the external bus. It specifies the physical address of the bus transaction. These signals can change during a transaction when controlled by the memory controller.
			I	Sampled by the MPC823 when an external device initiates a transaction and the memory controller was configured to handle external master accesses.
RD/$\overline{WR}$	1	High	O	**Read/Write**—Driven by the MPC823 along with the address when it owns the external bus. Driven high indicates that a read access is in progress and driven low indicates that a write access is in progress.
			I	Sampled by the MPC823 when an external device initiates a transaction and the memory controller was configured to handle external master accesses.
$\overline{BURST}$	1	Low	O	**Burst Transfer**—Driven by the MPC823 along with the address when it “owns” the external bus. Driven low indicates that a burst transfer is in progress and driven high indicates that the current transfer is not a burst.
			I	Sampled by the MPC823 when an external device initiates a transaction and the memory controller was configured to handle external master accesses.
TSIZ[0:1]	2	High	O	**Transfer Size**—Driven by the MPC823 along with the address when it owns the external bus. It specifies the data transfer size for the transaction.
			I	Sampled by the MPC823 when an external device initiates a transaction and the memory controller was configured to handle external master accesses.
AT[0:3]	3	High	O	**Address Type**—Driven by the MPC823 along with the address when it owns the external bus. It provides additional information about the address on the current transaction.
			I	Used only for testing purposes.
$\overline{RSV}$	1	Low	O	**Reservation Transfer**—Driven by the MPC823 along with the address when it owns the external bus. It provides additional information about the address on the current transaction.
			I	Used only for testing purposes.
$\overline{PTR}$	1	Low	O	**Program Trace**—Driven by the MPC823 along with the address when it owns the external bus. It provides additional information about the address on the current transaction.
			I	Used only for testing purposes.

Table 13-1. Bus Interface Signals (Continued)

MNEMONIC	PINS	ACTIVE	I/O	DESCRIPTION
$\overline{\text{BDIP}}$	1	Low	O	**Burst Data In Progress**—Driven by the MPC823 when it owns the external bus. It is part of the burst protocol. Asserted indicates that the second beat in front of the current one is requested by the master. This signal is negated prior to the end of a burst to terminate the burst data phase early.
			I	Used only for testing purposes.
TRANSFER START				
$\overline{\text{TS}}$	1	Low	O	**Transfer Start**—Driven by the MPC823 when it owns the external bus.It indicates the start of a transaction on the external bus.
			I	Sampled by the MPC823 when an external device initiates a transaction and the memory controller was configured to handle external master accesses.
$\overline{\text{STS}}$	1	Low	O	**Special Transfer Start**—Driven by the MPC823 when it owns the external bus. It indicates the start of a transaction on the external bus or an internal transaction in show cycle mode.
RESERVATION PROTOCOL				
$\overline{\text{KR}}$/$\overline{\text{RETRY}}$	1	Low	I	**Kill Reservation/Retry**—When a bus cycle is initiated by a **stwcx** instruction that was issued by the core to a nonlocal bus on which the storage reservation has been lost, this signal is used by the nonlocal bus interface to back-off the cycle. Refer to **Section 13.4.10 Storage Reservation Protocol**. For a regular transaction, this signal is driven by the slave device to indicate that the MPC823 has to relinquish ownership of the bus and retry the cycle.
DATA				
D[0:31]	32	High	I/O	**Data Bus**—The data bus has the following byte lane assignments: Data Byte — Byte Lane D(0:7) — 0 D(8:15) — 1 D(16:23) — 2 D(24:31) — 3
			O	Driven by the MPC823 when it owns the external bus and has initiated a write transaction to a slave device. For single beat transactions, if external A(6:31) and TSIZ(0:1) do not select the byte lanes for transfer, they will not supply valid data.
			I	Driven by the slave in a read transaction. For single beat transactions, if external A(6:31) and TSIZ(0:1) do not select the byte lanes for transfer, they will not be sampled by the MPC823. It is also sampled by the MPC823 when the external master acquires the bus.

Table 13-1. Bus Interface Signals (Continued)

MNEMONIC	PINS	ACTIVE	I/O	DESCRIPTION
DP[0:3]	4	High	I/O	**Parity Bus**—Each parity signal corresponds to each one of the data bus lanes: Data Bus Byte — Parity Line D(0:7) — DP0 D(8:15) — DP1 D(16:23) — DP2 D(24:31) — DP3
			O	Driven by the MPC823 when it owns the external bus and has initiated a write transaction to a slave device. Each parity signal has the parity value (even or odd) of the corresponding data bus byte. For single beat transactions, if external A(6:31) and TSIZ(0:1) do not select the byte lanes for transfer, they will not have a valid parity line.
			I	Driven by the slave in a read transaction. Each parity signal is sampled by the MPC823 and checked (if enabled) against the expected value parity value (even or odd) of the corresponding data bus byte. For single beat transactions, if external A(6:31) and TSIZ(0:1) do not select the byte lanes for transfer, they will not be sampled by the MPC823 and its parity signals will not be checked.
TRANSFER CYCLE TERMINATION				
$\overline{TA}$	1	Low	I	**Transfer Acknowledge**—Driven by the slave device the current transaction was addressed to. It indicates that the slave has received the data on the write cycle or returned the data on the read cycle. If the transaction is a burst, $\overline{TA}$ should be asserted for each one of the transaction beats.
			O	Driven by the MPC823 when the slave device is controlled by the on-chip memory controller.
$\overline{TEA}$	1	Low	I	**Transfer Error Acknowledge**—Driven by the slave device the current transaction was addressed to. It indicates that an error condition has occurred during the bus cycle.
			O	Driven by the MPC823 when the internal bus monitor detects an erroneous bus condition.
$\overline{BI}$	1	Low	I	**Burst Inhibit**—Driven by the slave device the current transaction was addressed to. It indicates that the current slave does not support burst mode.
			O	Driven by the MPC823 when the slave device is controlled by the on-chip memory controller.

Table 13-1. Bus Interface Signals (Continued)

MNEMONIC	PINS	ACTIVE	I/O	DESCRIPTION
ARBITRATION				
$\overline{BR}$	1	Low	I	**Bus Request**—When the internal arbiter is asserted, it indicates that an external master is requesting the bus.
			O	Driven by the MPC823 when the internal arbiter is disabled and the chip is not parked.
$\overline{BG}$	1	Low	O	**Bus Grant**—When the internal arbiter is enabled, the MPC823 asserts this signal to indicate that an external master can assume ownership of the bus and begin a bus transaction. The $\overline{BG}$ signal should be qualified by the master requesting the bus to ensure it is the bus owner: Qualified $\overline{BG}$ = $\overline{BG}$ & ~ $\overline{BB}$
			I	When the internal arbiter is disabled, the $\overline{BG}$ is sampled and properly qualified by the MPC823 when an external bus transaction is to be executed by the chip.
$\overline{BB}$	1	Low	O	**Bus Busy**—When the internal arbiter is enabled, the MPC823 asserts this signal to indicate that it is the current owner of the bus. When the internal arbiter is disabled, it will assert this signal after the external arbiter grants the chip ownership of the bus and it is ready to start the transaction.
			I	When the internal arbiter is enabled, the MPC823 samples this signal to get an indication of when the external master ended its bus tenure ($\overline{BB}$ negated). When the internal arbiter is disabled, the $\overline{BB}$ is sampled to properly qualify the $\overline{BG}$ line when an external bus transaction is to be executed by the chip.

NOTE: O indicates an output from the MPC823 and I indicates an input.

13.4 BUS INTERFACE OPERATION

The MPC823 generates a system clock output (CLKOUT) that sets the frequency of operation for the bus interface. Internally, the MPC823 uses a phase-lock loop (PLL) circuit to generate a master clock for all of the core circuitry, which is phase-locked to the CLKOUT output signal.

All signals for the MPC823 bus interface are specified with respect to the rising-edge of the external CLKOUT and are guaranteed to be sampled as inputs or changed as outputs with respect to that edge. Since the same clock edge is referenced for driving or sampling the bus signals, the possibility of clock skew could exist between various modules in a system because of routing or using multiple clock lines. It is your responsibility to handle any clock skew problems that could occur as a result of layout, lead-length, and physical routing.

13.4.1 Basic Transfers

The basic transfer protocol defines the sequence of actions that must occur on the MPC823 bus to perform a complete bus transaction. The chronological sequence or phase of a typical bus transfer is as follows:

1. Arbitration
2. Address transfer
3. Data transfer
4. Termination

This protocol provides for an arbitration phase and an address and data transfer phase. The arbitration phase specifies the master that initiates the next transaction. The address phase specifies the address for the transaction and the transfer attributes that describe the transaction. The data phase performs the transfer of data. It can transfer a single beat of data (4 bytes or less) for nonburst operations, a 4-beat burst of data, an 8-beat burst of data, or a 16-beat burst of data.

13.4.2 Single Beat Transfers

During the data transfer phase, data is transferred from master to slave on write cycles or from slave to master on read cycles. On a write cycle, the master drives the data as soon as it can, but never before the cycle following the address transfer phase. The master has to take into consideration the “one dead clock cycle” when switching between drivers to avoid electrical contention. The master can stop driving the data bus as soon as it samples the $\overline{TA}$ line asserted on the rising edge of the CLKOUT. On a read cycle the master accepts the data bus contents as valid at the rising edge of the CLKOUT in which the $\overline{TA}$ signal is sampled asserted.

13.4.2.1 SINGLE BEAT READ FLOW. The basic read cycle begins with a bus arbitration, followed by the address transfer and the data transfer. The handshakes are illustrated in the following diagrams as applicable to the fixed transaction protocol.

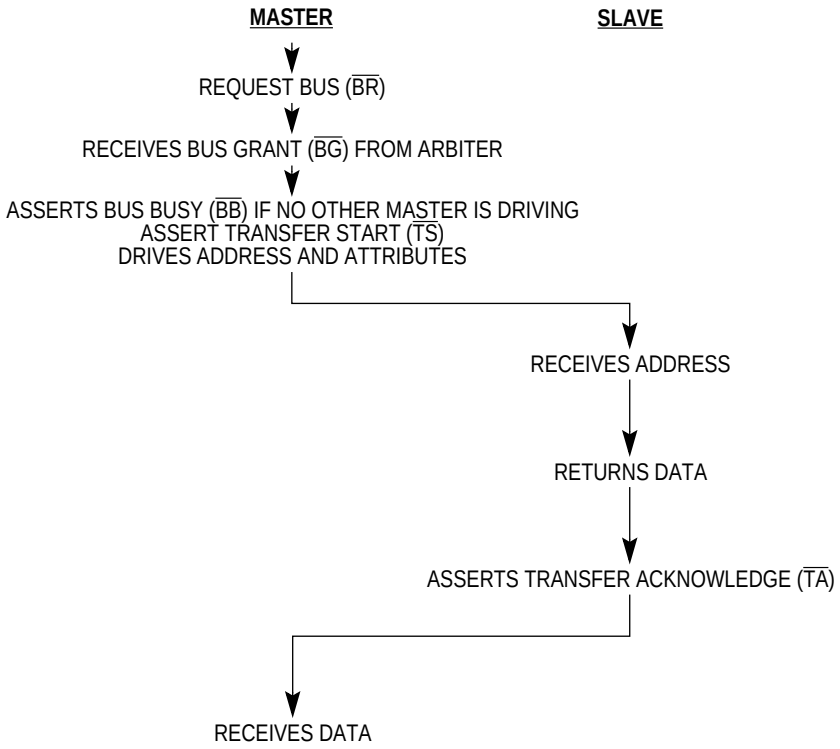

Figure 13-3. Basic Flow Diagram of a Single Beat Read Cycle

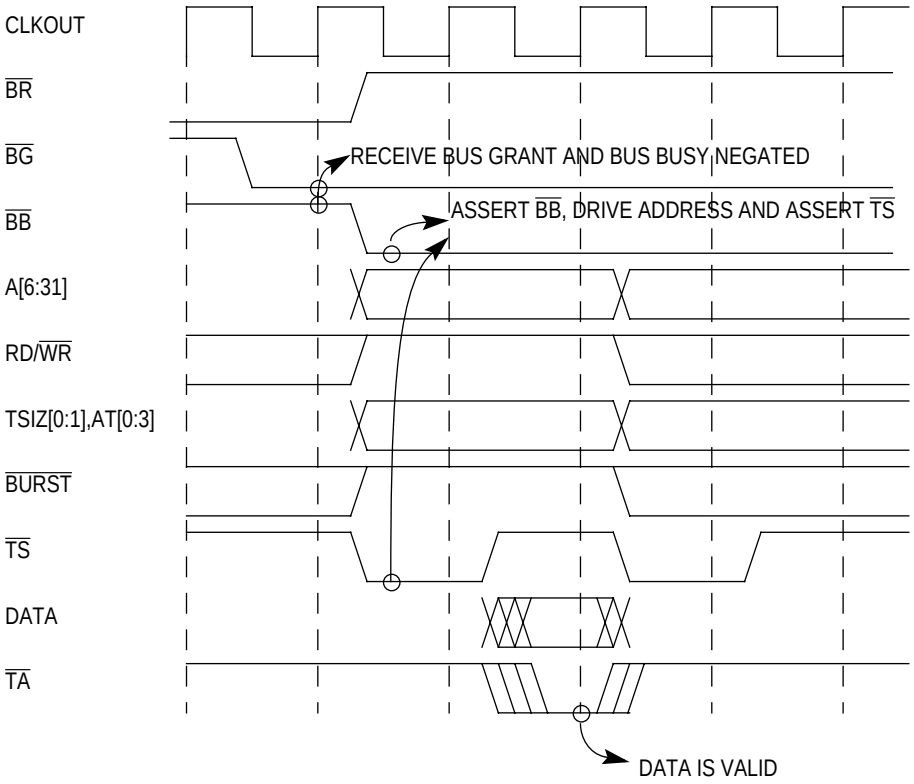

Figure 13-4. Single Beat Read Cycle–Basic Timing–Zero Wait States

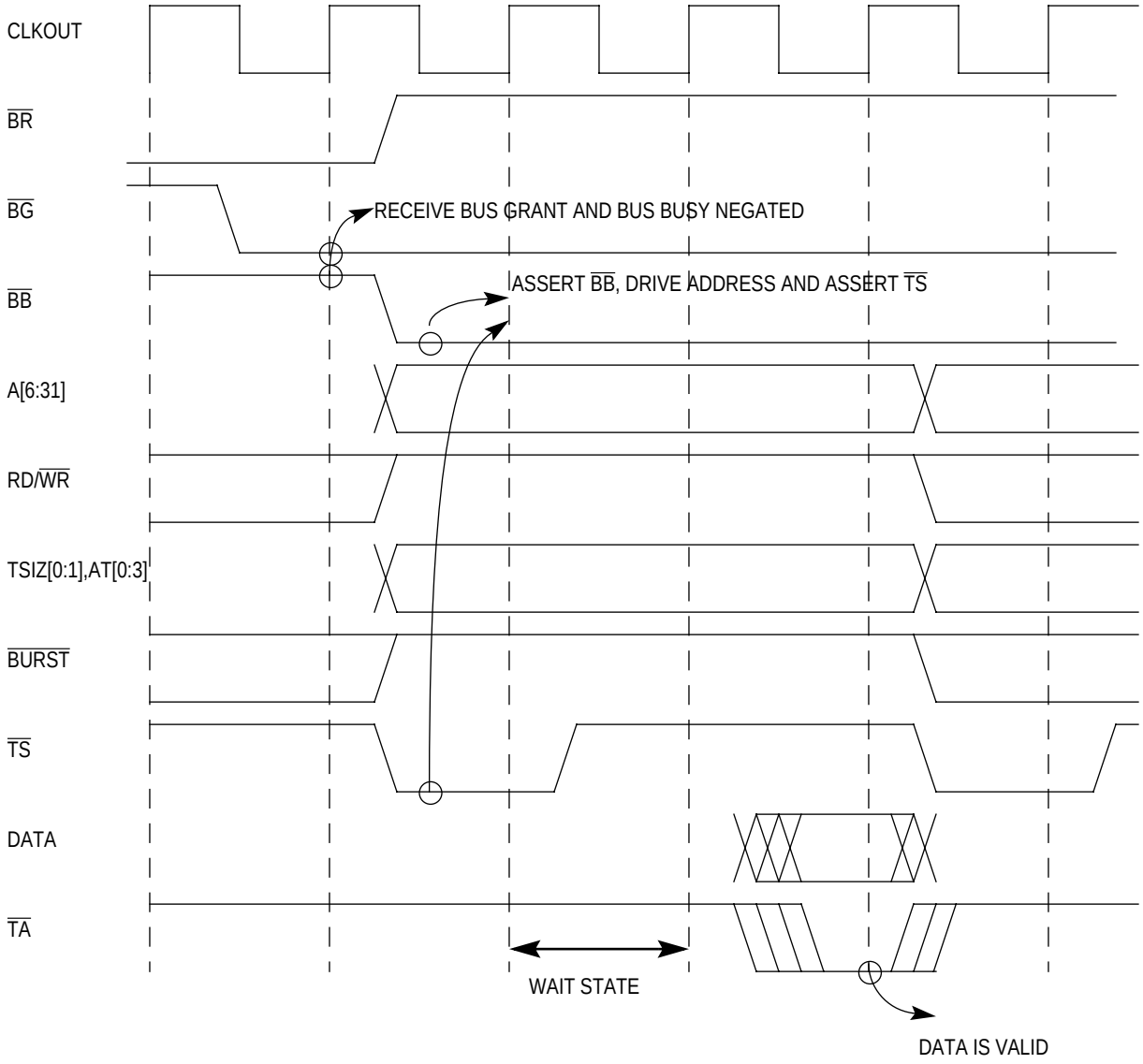

Figure 13-5. Single Beat Read Cycle–Basic Timing–One Wait State

13.4.2.2 SINGLE BEAT WRITE FLOW. The basic write cycle begins with a bus arbitration, followed by the address transfer and the data transfer. The handshakes are illustrated in Figure 13-6, Figure 13-7, Figure 13-8, and Figure 13-9 as applicable to the fixed transaction protocol.

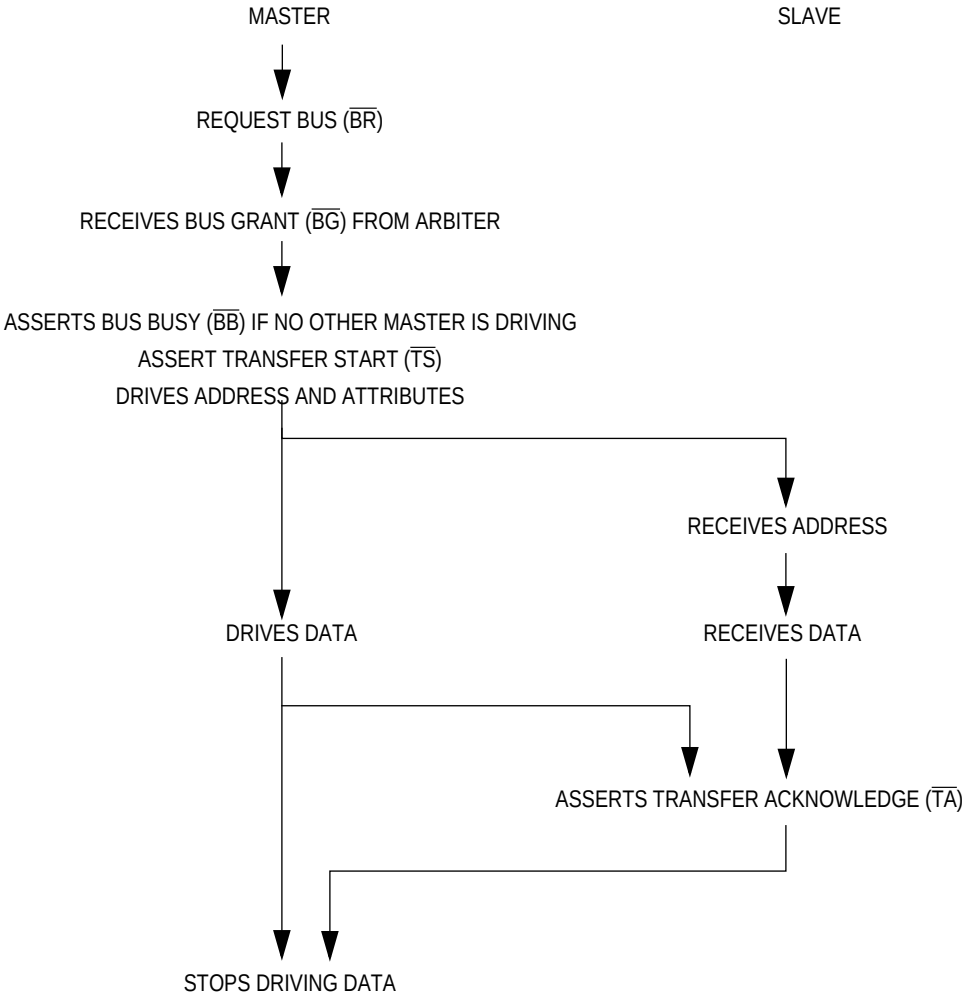

Figure 13-6. Basic Flow Diagram of a Single Beat Write Cycle

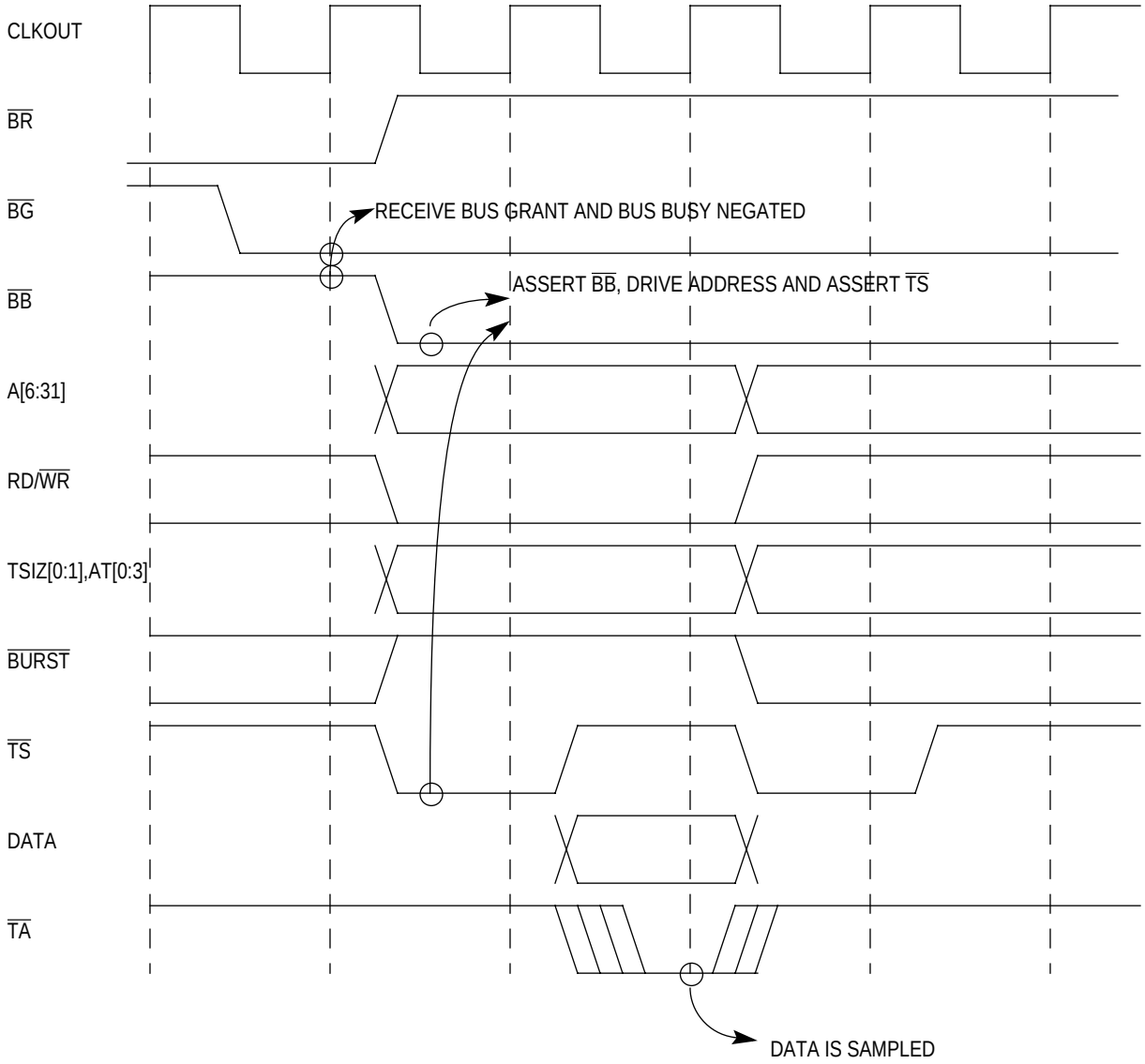

Figure 13-7. Single Beat Write Cycle–Basic Timing–Zero Wait States

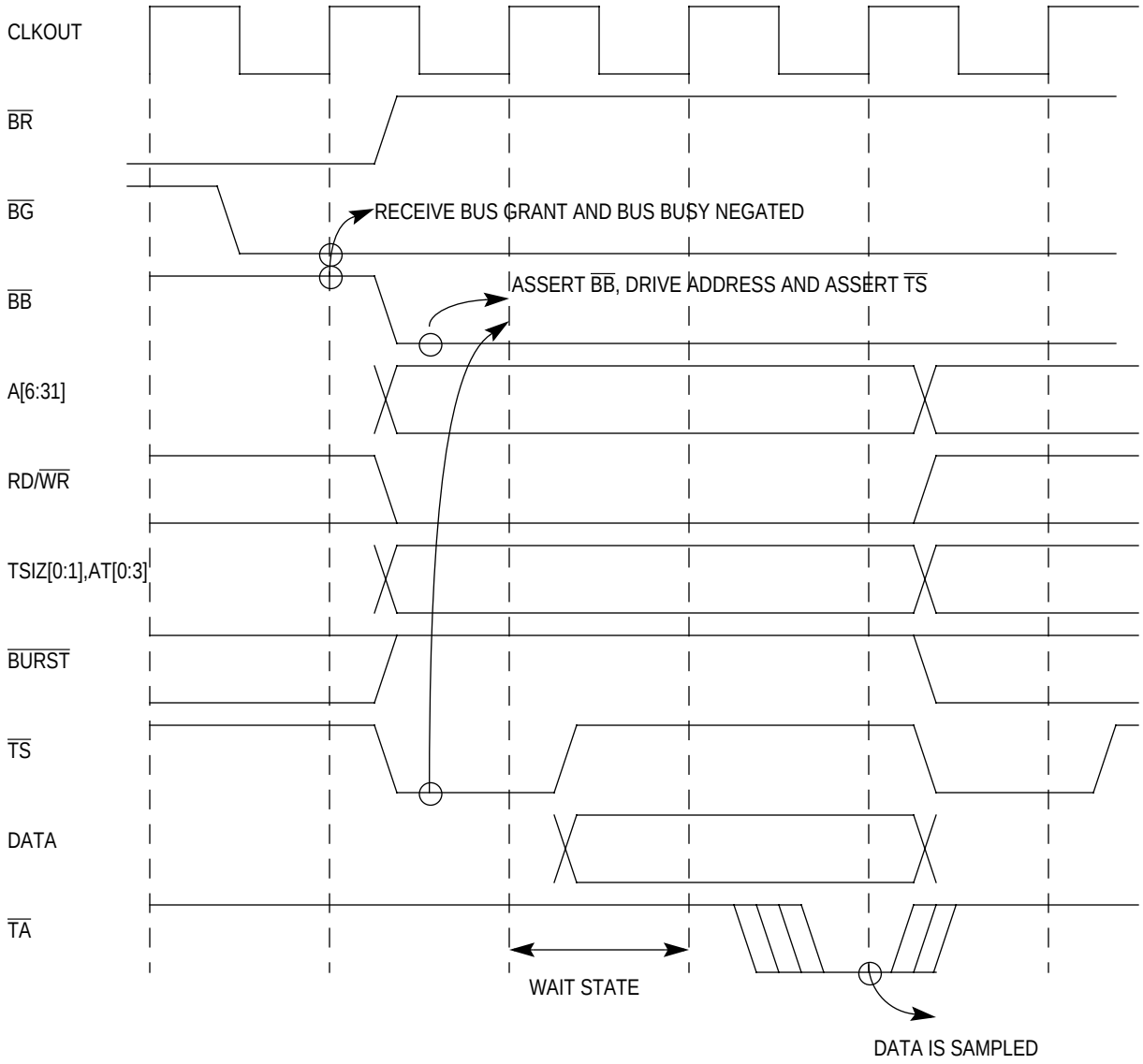

Figure 13-8. Single Beat Write Cycle of One Wait State

A typical single beat transfer assumes that the external memory has a 32-bit port size. The MPC823 provides an effective mechanism for interfacing with 16-bit port size memories and 8-bit port size memories, thus allowing transfers to these devices when they are controlled by the internal memory controller. The port size (PS) timing shown in the following figures is representative of the PS field in **Section 15 Memory Controller**.

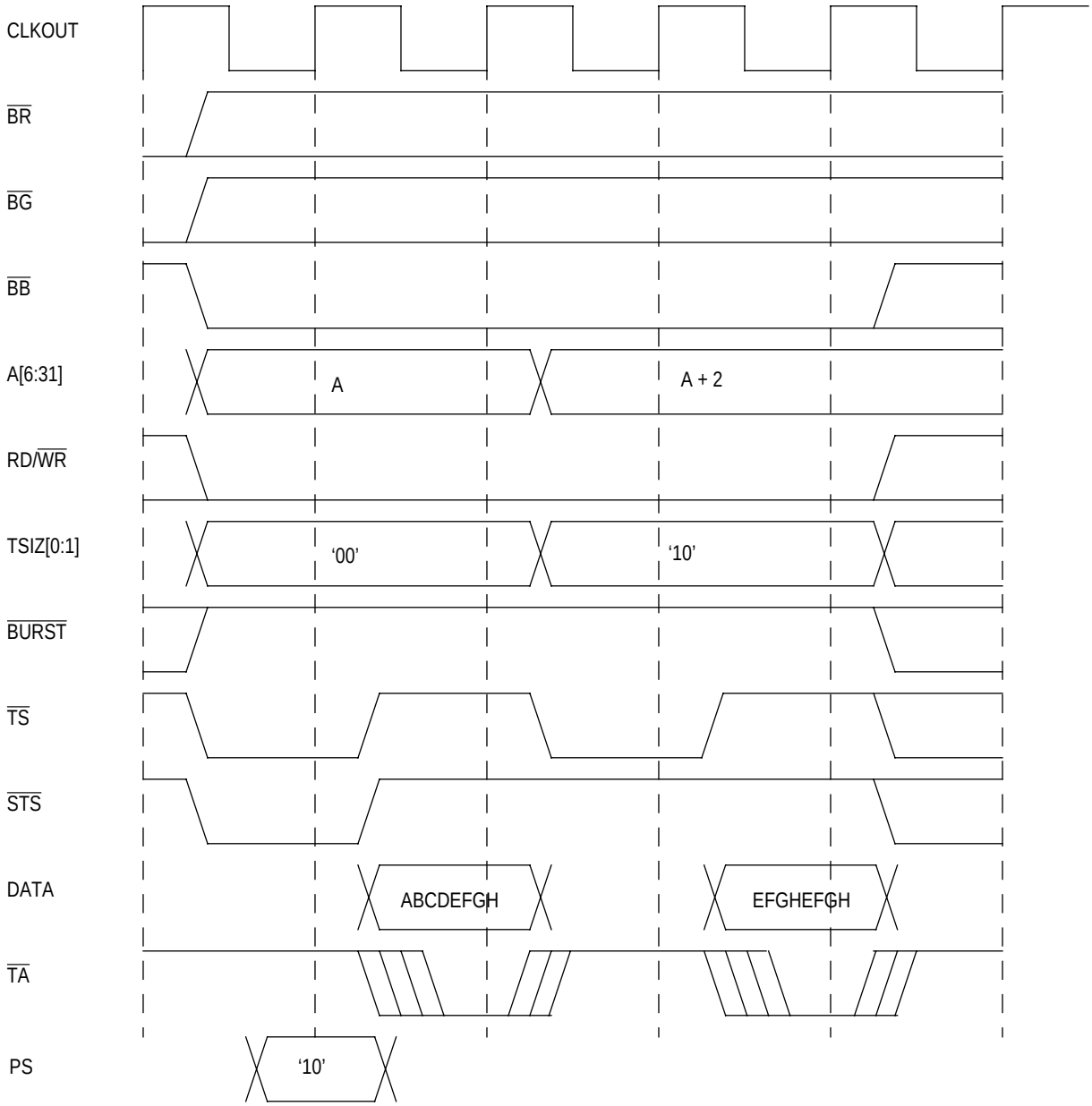

Figure 13-9. Single Beat, 32-Bit Data, Write Cycle From a 16-Bit Port Size

13.4.3 Burst Transfers

The MPC823 uses burst transfers to access 16-byte operands. A burst accesses a block of 16 bytes that must be aligned to a 16-byte memory boundary by supplying a starting address that points to the critical words and requiring the memory device to sequentially drive/sample each word on the data bus. The selected slave device must internally increment the external A[28:29] signal (or A[30] for a 16-bit port size slave device) of the supplied address for each transfer, thus causing the address to wrap around at the end of the 4-word block. The address and transfer attributes supplied by the MPC823 remain stable during the transfers and the selected device terminates each transfer by driving/sampling the word on the data bus and asserting the $\overline{TA}$ signal.

The MPC823 also supports burst-inhibited transfers for slave devices that are unable to support bursting. For this type of bus cycle, the selected slave device supplies/samples the first word the MPC823 points to and asserts the $\overline{BI}$ signal with $\overline{TA}$ for the first transfer of the burst access. The MPC823 responds by terminating the burst and accessing the remainder of the 16-byte block, thus using three read/write cycle bus (each one for a word) for a 32-bit port width slave, seven read/write cycle bus for a 16-bit port width slave, or fifteen read/write cycle bus for an 8-bit port width slave.

Burst transfers assume that the external memory has a 32-bit port size. The MPC823 provides an effective mechanism for interfacing with 16- and 8-bit port size memories that allow burst transfers to these devices when they are controlled by the internal memory controller. The MPC823 attempts to initiate a burst transfer as normal. If the slave device responds to a cycle prior to the $\overline{TA}$ signal for the first beat, its port size is 8 or 16 bits and the MPC823 completes a burst of 8- or 16-bit beats. Effectively, each of the data beats of the burst transfers only 1 or 2 bytes. This 8- or 16-beat burst is also considered an atomic transaction, so the MPC823 will not allow other unrelated master accesses or bus arbitration to intervene between the transfers.

13.4.4 The Burst Mechanism

The MPC823 burst mechanism consists of one signal indicating that the cycle is a burst cycle, one indicating the duration of the burst data, and another signal indicating whether the slave is burstable. These signals are in addition to the basic signals of the bus. At the start of the burst transfer, the master drives the address, address attributes, and $\overline{BURST}$ signal to indicate that a burst transfer is being initiated, along with the assertion of the $\overline{TS}$ signal. If the slave is burstable, it negates the $\overline{BI}$ signal. If the slave cannot burst, it must assert the $\overline{BI}$ signal. During the data phase of a burst write cycle the master drives the data. It also asserts the $\overline{BDIP}$ signal if it intends to drive a subsequent data beat after the current data beat. When the slave has received the data, it asserts the $\overline{TA}$ signal to let the master know it is ready for the next data transfer. The master again drives the next data and asserts or negates the $\overline{BDIP}$ signal. If the master does not intend to drive another data beat after the current one, it negates the $\overline{BDIP}$ signal to let the slave know that the next subsequent data beat transfer is the last data of the burst write transfer. During the data phase of a burst read cycle, the master receives data from the addressed slave. If the master needs more than one data, it asserts the $\overline{BDIP}$ signal. When the data is received prior to the last data, the master negates the $\overline{BDIP}$ signal. Thus, the slave stops driving new data after it receives the negation of the $\overline{BDIP}$ signal at the rising edge of the clock. See Figure 13-10 for details.

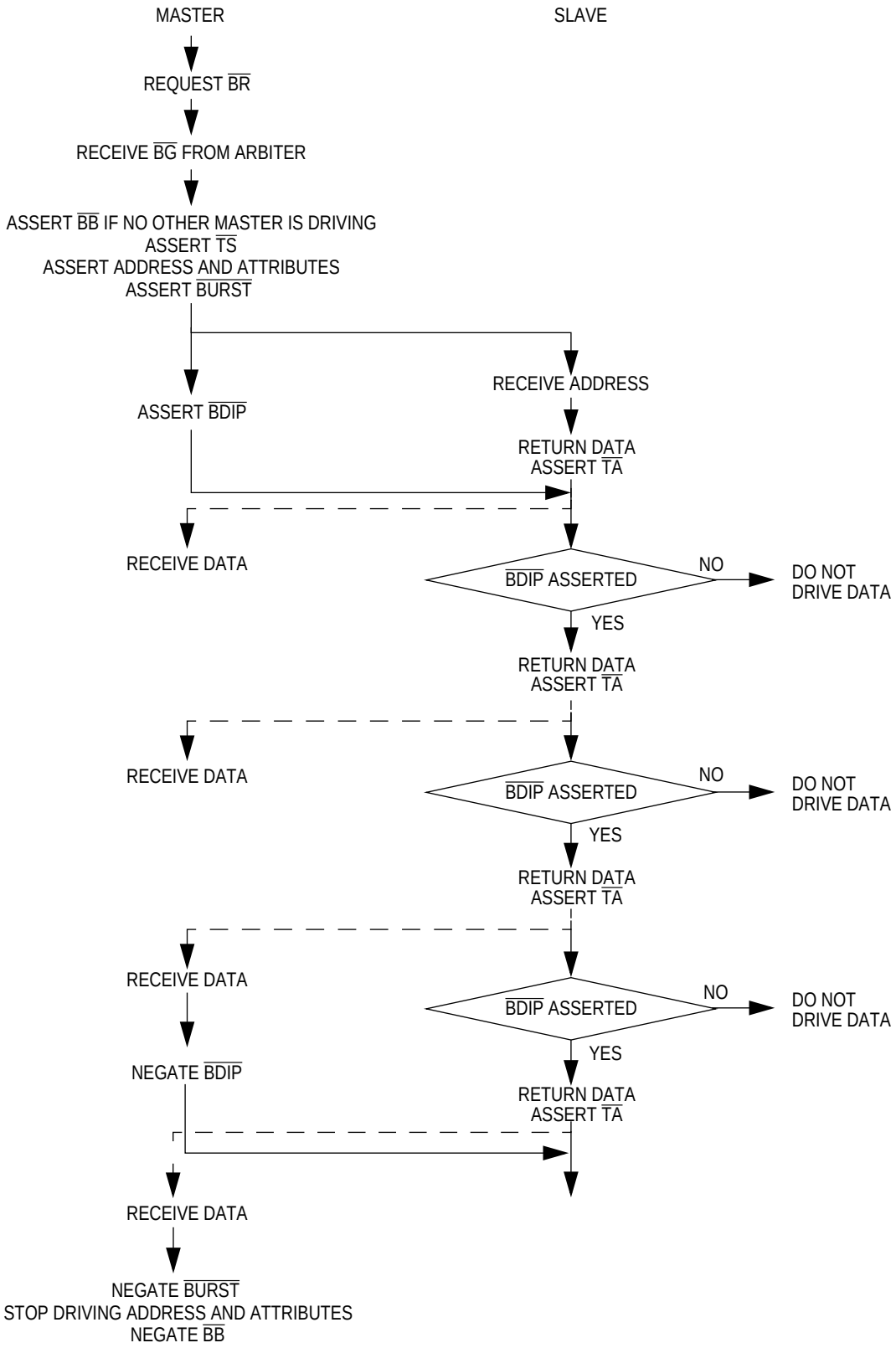

Figure 13-10. Basic Flow Diagram Of A Burst Read Cycle

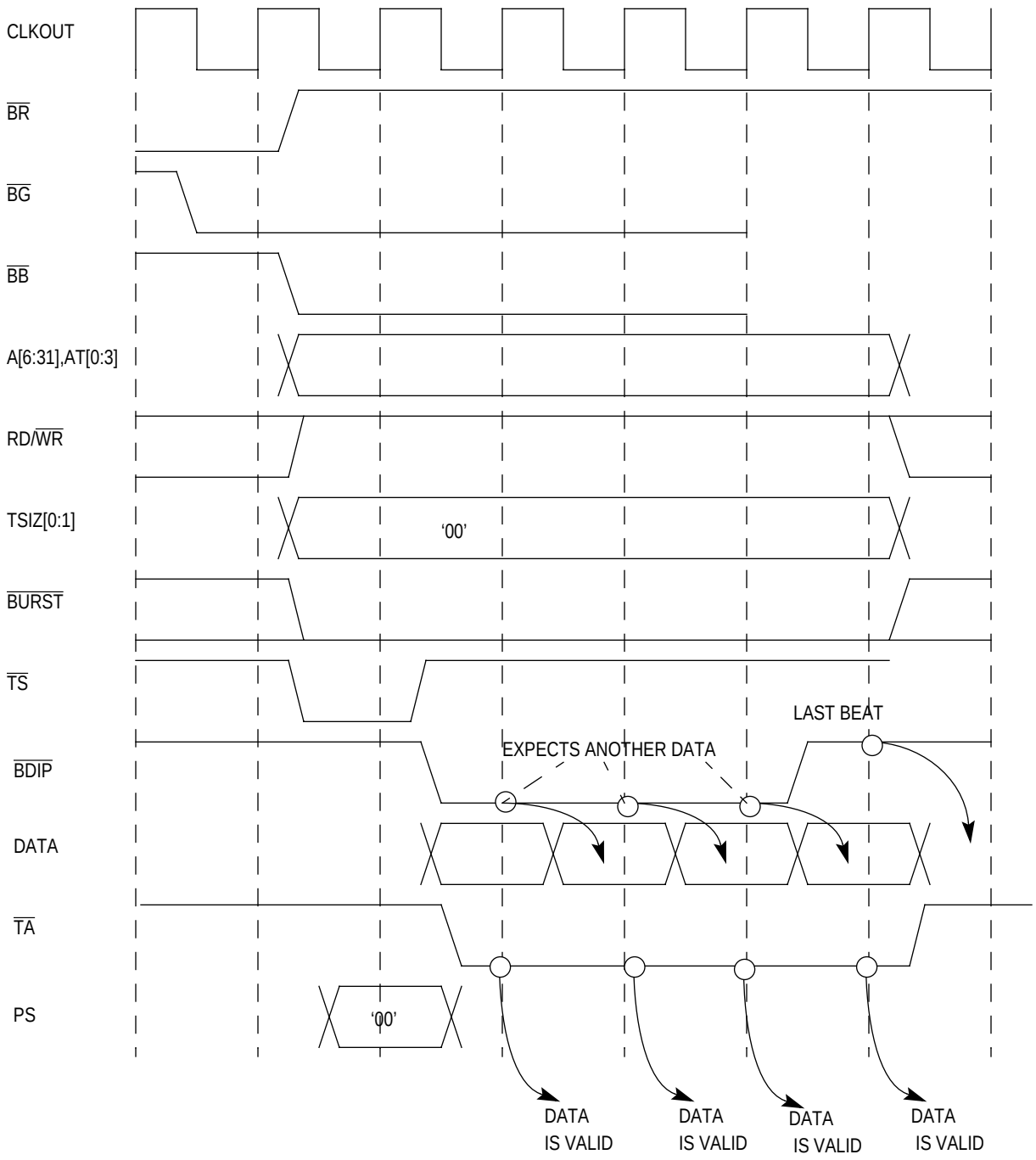

Figure 13-11. Burst-Read Cycle–32-Bit Port Size–Zero Wait State

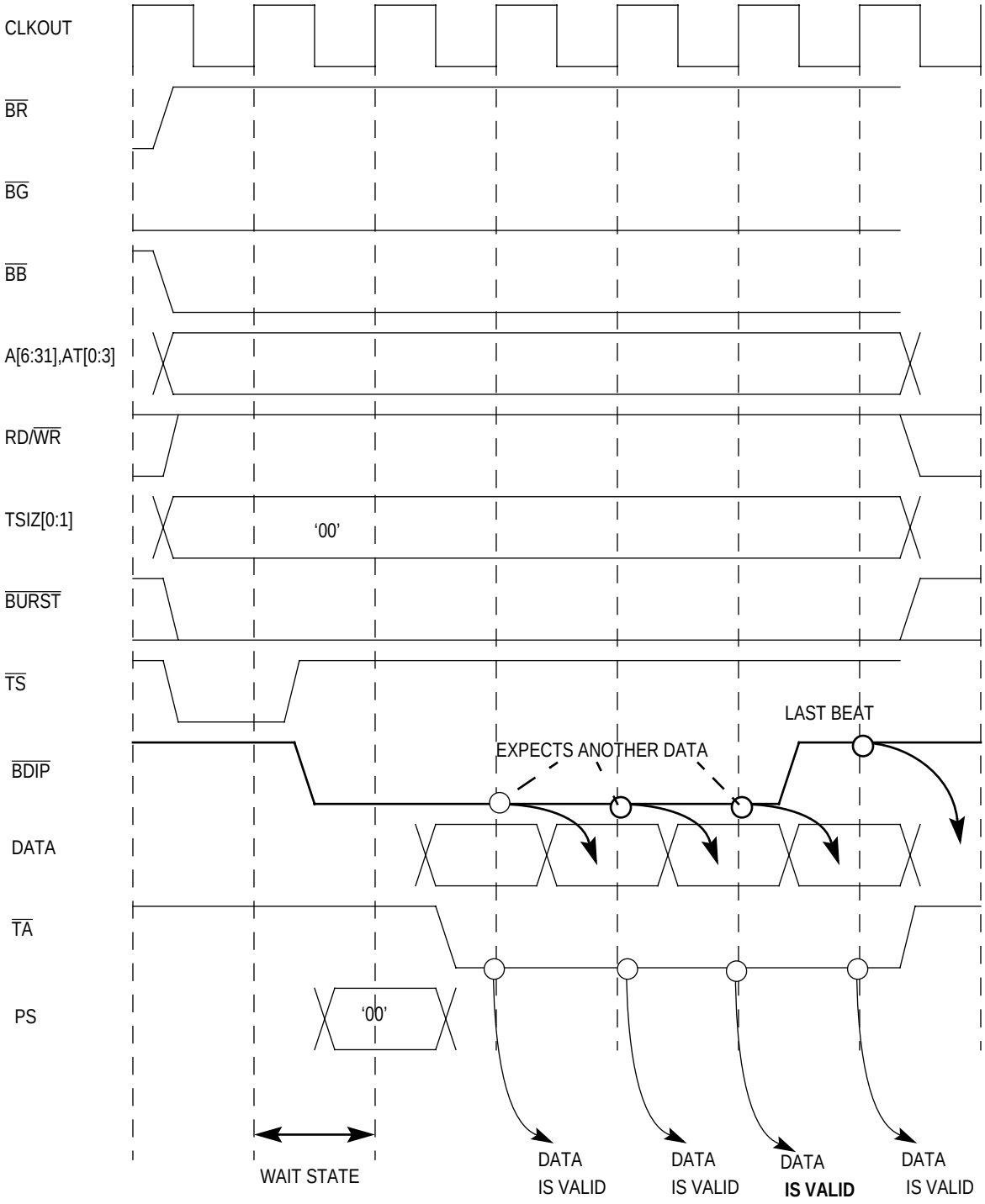

Figure 13-12. Burst-Read Cycle–32-Bit Port Size–One Wait State

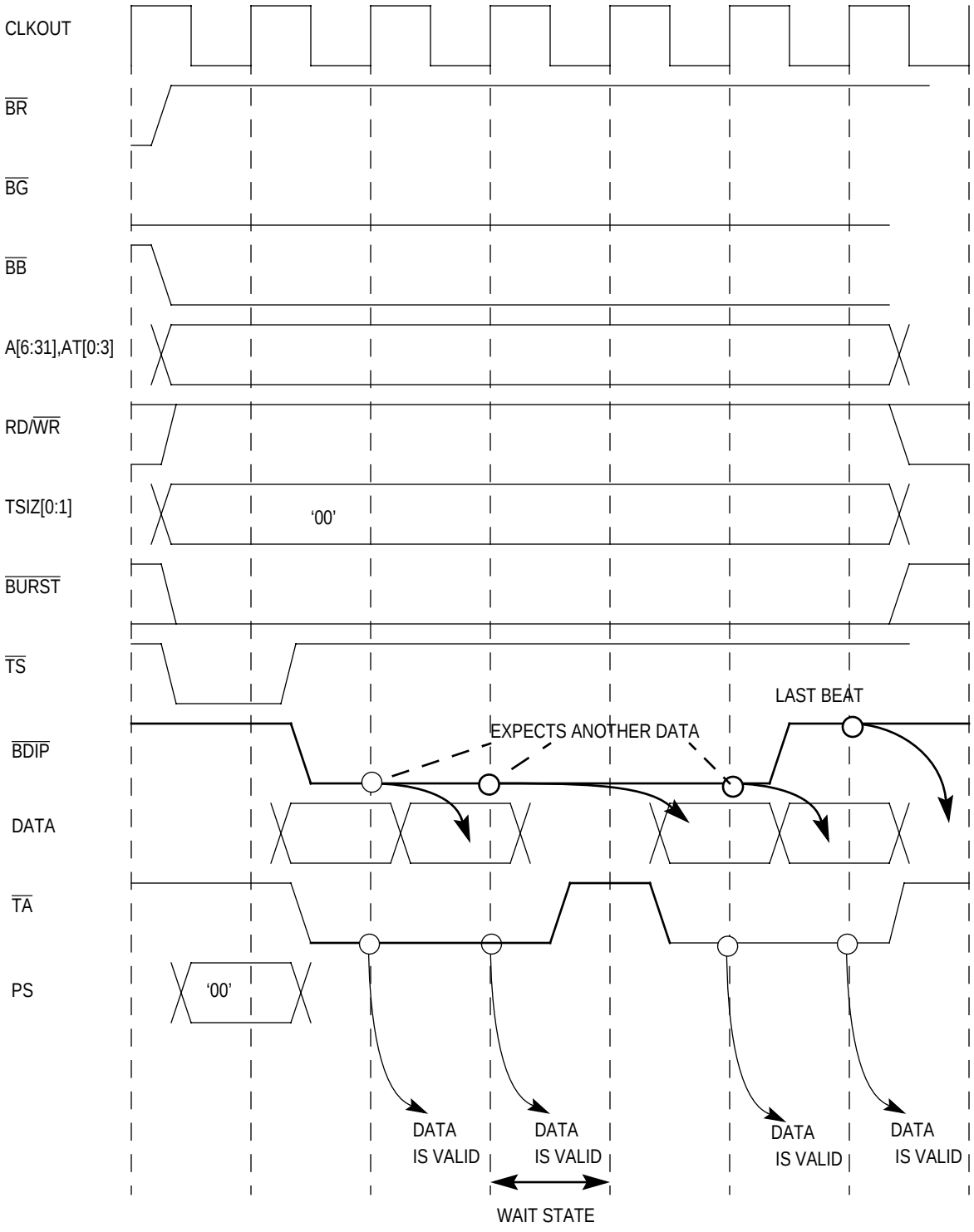

Figure 13-13. Burst-Read Cycle–32-Bit Port Size–Wait States Between Beats

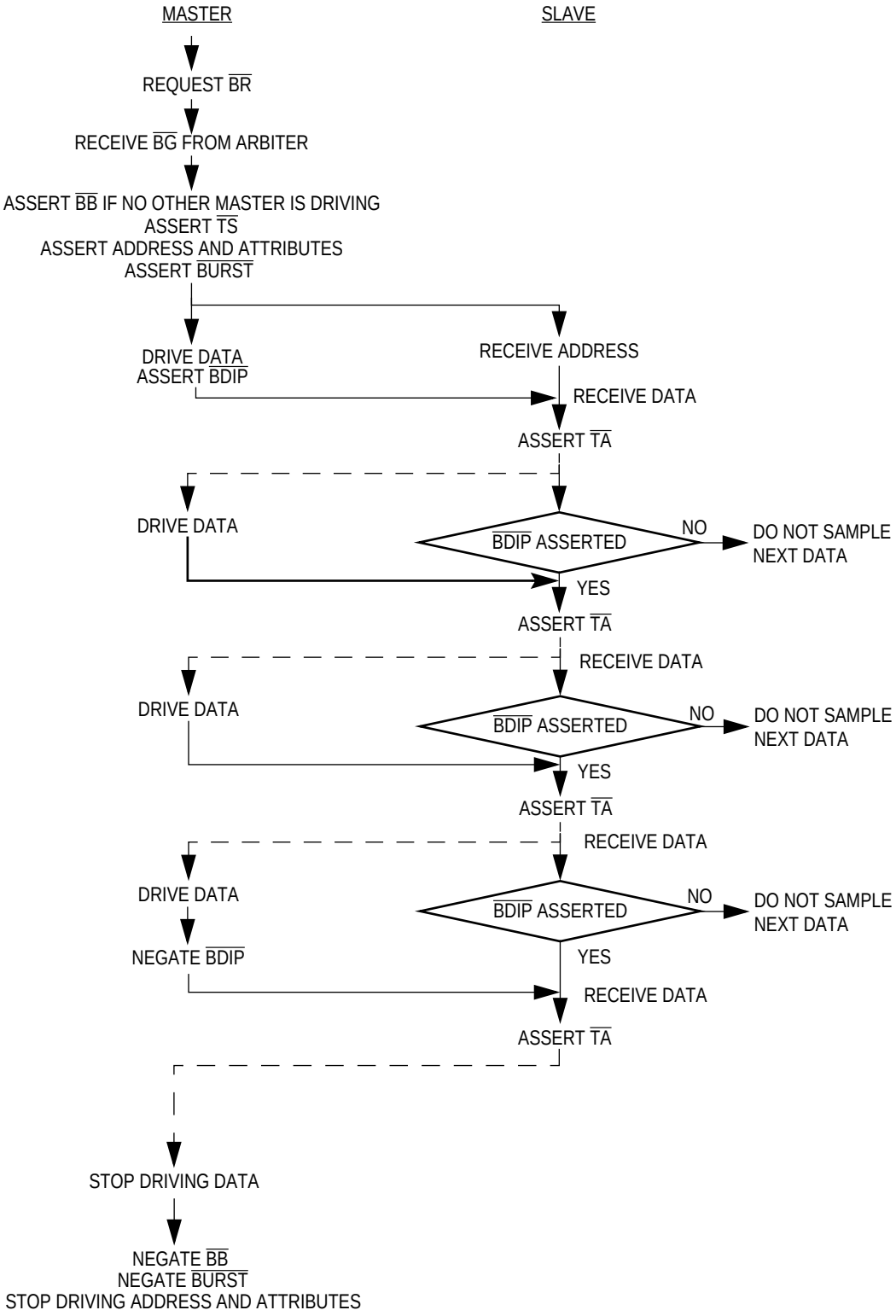

Figure 13-14. Basic Flow Diagram of a Burst Write Cycle

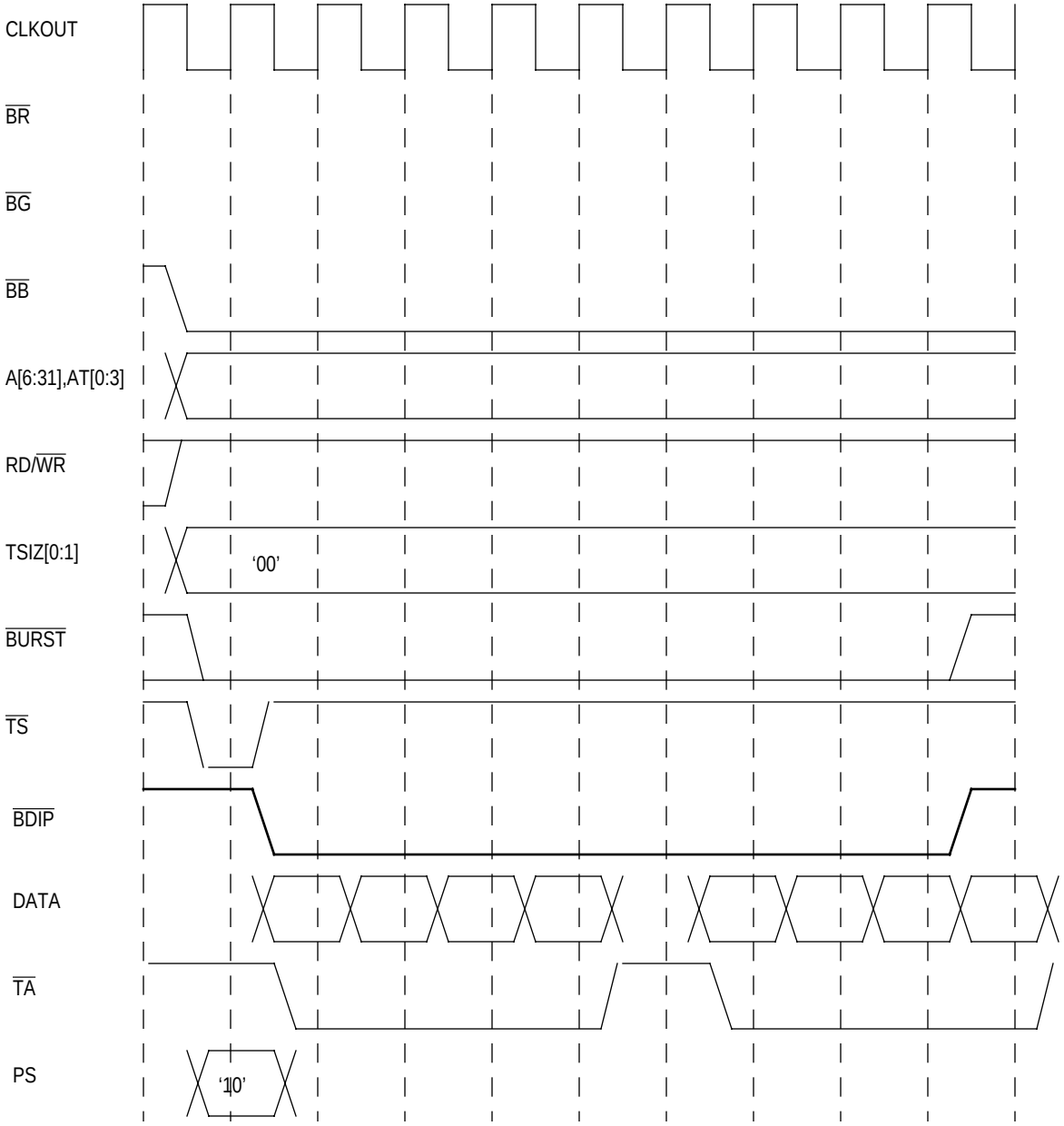

Figure 13-15. Burst-Read Cycle–16-Bit Port Size–One Wait State Between Beats

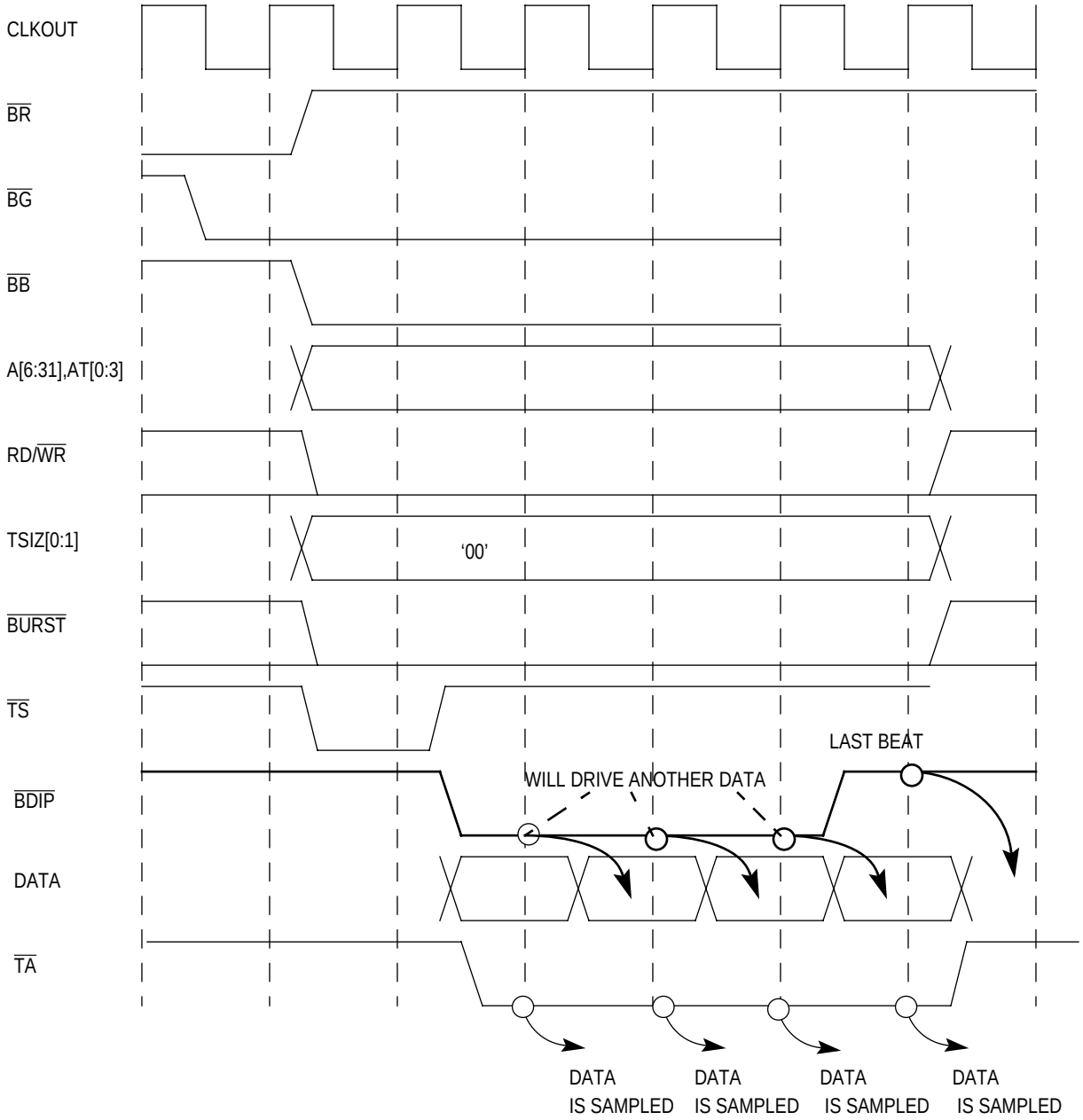

Figure 13-16. Burst-Write Cycle–32-Bit Port Size–Zero Wait States

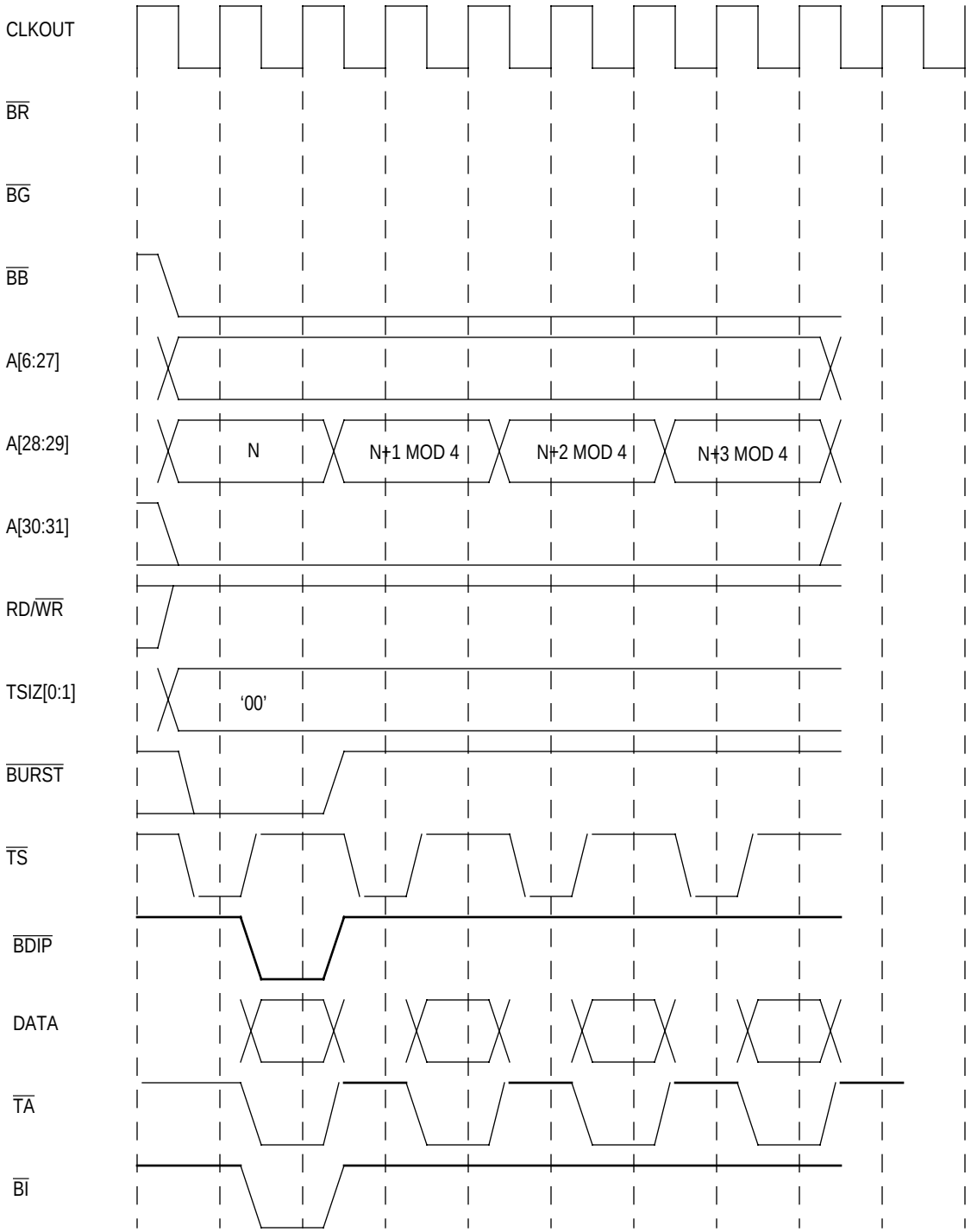

Figure 13-17. Burst-Inhibit Cycle–32-Bit Port Size

13.4.5 Transfer Alignment and Packaging

The MPC823 external bus only supports natural address alignment that forces the following restrictions:

- Byte access can have any address alignment
- Half-word access must have address bit 31 equal to 0
- Word access must have address 30-31 equal to 0
- For burst access must have address 30-31 equal to 0

The MPC823 can perform operand transfers through its 32-bit data port. If the transfer is controlled by the internal memory controller, the MPC823 can support 8- and 16-bit data port sizes. The bus requires that the portion of the data bus used for a transfer to or from a particular port size to be fixed. A 32-bit port must reside on data bus bits 0-31, a 16-bit port must reside on bits 0-15, and an 8-bit port must reside on bits 0-7. The MPC823 always tries to transfer the maximum amount of data on all bus cycles and for a word operation it always assumes that the port is 32 bits wide at the beginning of the bus cycle. In Figure 13-18 and Figure 13-19 and Table 13-2 and Table 13-3, the following operand conventions have been adopted:

- OP0 is the most-significant byte of a word operand and OP3 is the least-significant byte.
- The two bytes of a half-word operand are OP0 (most-significant) and OP1 or OP2 (least-significant) and OP3, depending on the address of the access.
- The single byte of a byte-length operand is OP0, OP1, OP2, or OP3, depending on the address of the access.

Note: Although this is a 32-bit machine, only 26 of the bits are visible outside the chip.

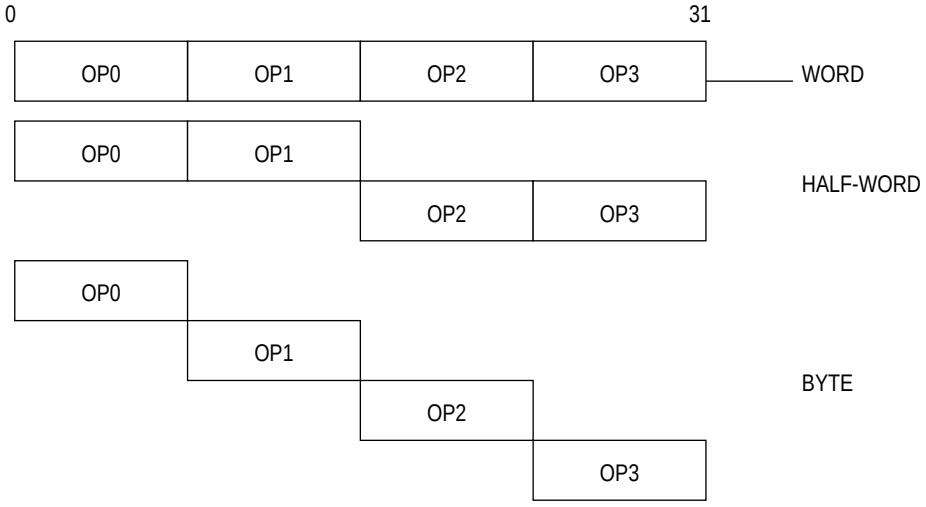

Figure 13-18. Internal Operand Representation

Figure 13-19 illustrates the device connections on the data bus.

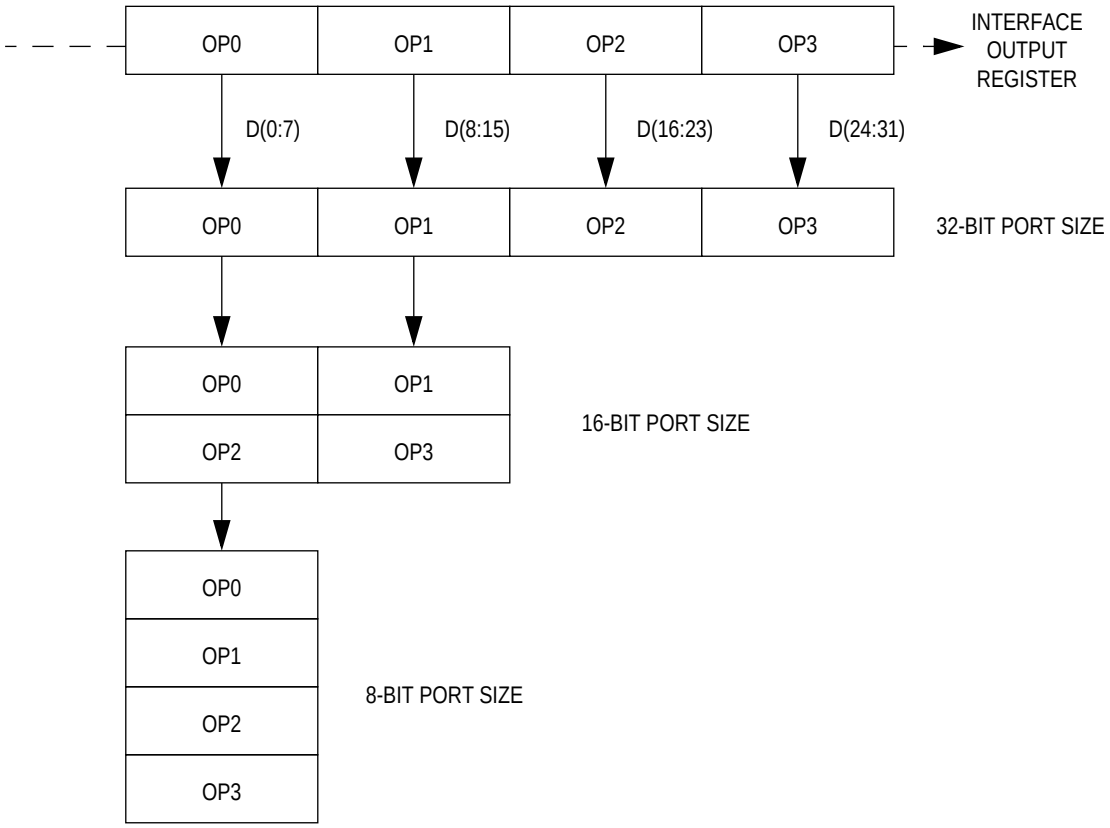

Figure 13-19. Interface To Different Port Size Devices

Table 13-2 lists the bytes for read cycles required on the data bus.

Table 13-2. Data Bus Requirements For Read Cycles

TRANSFER SIZE	TSIZE [0:1]		INTERNAL ADDRESS		32-BIT PORT SIZE				16-BIT PORT SIZE		8-BIT PORT SIZE
			A30	A31	D0–D7	D8–D15	D16–D23	D24–D31	D0–D7	D8–D15	D0–D7
Byte	0	1	0	0	OP0	—	—	—	OP0	—	OP0
	0	1	0	1	—	OP1	—	—	—	OP1	OP1
	0	1	1	0	—	—	OP2	—	OP2	—	OP2
	0	1	1	1	—	—	—	OP3	—	OP3	OP3
Half-Word	1	0	0	0	OP0	OP1	—	—	OP0	OP1	OP0
	1	0	1	0	—	—	OP2	OP3	OP2	OP3	OP2
Word	0	0	0	0	OP0	OP1	OP2	OP3	OP0	OP1	OP0

NOTE: — Denotes that a byte is not required during that read cycle.

Table 13-3 lists the patterns of the data transfer for write cycles when accesses are initiated by the MPC823.

Table 13-3. Data Bus Contents for Write Cycles

TRANSFER SIZE	TSIZE [0:1]		INTERNAL ADDRESS		EXTERNAL DATA BUS PATTERN			
			A30	A31	D0–D7	D8–D15	D16–D23	D24–D31
Byte	0	1	0	0	OP0	—	—	—
	0	1	0	1	OP1	OP1	—	—
	0	1	1	0	OP2	—	OP2	—
	0	1	1	1	OP3	OP3	—	OP3
Half-Word	1	0	0	0	OP0	OP1	—	—
	1	0	1	0	OP2	OP3	OP2	OP3
Word	0	0	0	0	OP0	OP1	OP2	OP3

NOTE: — Denotes that a byte is not required during that read cycle.

13.4.6 Arbitration Phase-Related Signals

The external bus design provides for a single bus master, either the MPC823 or an external device. One or more of the external devices on the bus has the capability of becoming bus master for the external bus. Bus arbitration may be handled either by an external central bus arbiter or by the internal on-chip arbiter. In the latter case, the system is optimized for one external bus master besides the MPC823. The arbitration configuration (external or internal) is set at system reset. See **Section 15.6 External Master Support** for more information.

Each bus master must have $\overline{BR}$, $\overline{BG}$, and $\overline{BB}$ signals. The device that needs the bus asserts the $\overline{BR}$ signal. The device then waits for the arbiter to assert the $\overline{BG}$ signal. In addition, the new master must look at the $\overline{BB}$ signal to ensure that no other master is driving the bus before it can assert $\overline{BB}$ and assume ownership of the bus. If the arbiter has taken the $\overline{BG}$ away from the master and the master wants to execute a new cycle, the master must rearbitrate before a new cycle can be initiated. The MPC823, however, guarantees data coherency for burst accesses to a small port size. This means that the MPC823 will not release the bus until the transactions (atomic) complete.

Figure 13-20 describes the basic protocol for bus arbitration. For more information, see **Section 12.12.1.1 SIU Module Configuration Register**.

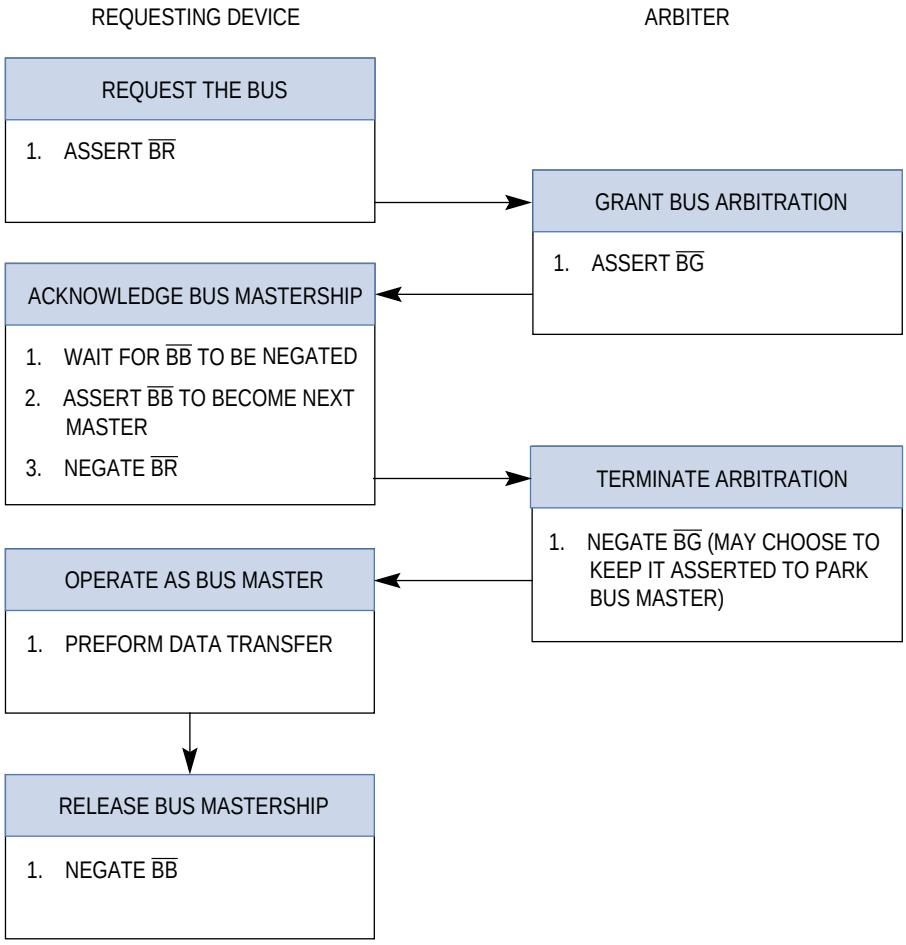

Figure 13-20. Bus Arbitration Flowchart

13.4.6.1 BUS REQUEST SIGNAL. The potential bus master asserts the $\overline{BR}$ signal to request bus mastership. $\overline{BR}$ should be negated once the bus is granted, the bus is not busy, and the new master can drive the bus. If more requests are pending, the master can keep asserting its bus request as long as needed. When configured for external central arbitration, the MPC823 drives this signal when it needs bus mastership. When the internal on-chip arbiter is used, this signal is an input to the internal arbiter and should be driven by the external bus master.

13.4.6.2 BUS GRANT SIGNAL. The $\overline{BG}$ signal is asserted by the arbiter to indicate that the bus is granted to the requesting device. The $\overline{BG}$ signal can be negated after $\overline{BR}$ is negated. The current bus master may choose to keep BG asserted to park the bus and maintain ownership without rearbitrating until another master makes a request. This reduces arbitration time, which then improves performance. When configured for external central arbitration, the $\overline{BG}$ becomes an input signal to the MPC823 from the external arbiter. When the internal on-chip arbiter is used, this signal is an output from the internal arbiter to the external bus master.

13.4.6.3 BUS BUSY SIGNAL. The $\overline{BB}$ signal indicates that the current bus master is using the bus. New masters should not begin transferring until this signal is negated. The bus owner should not relinquish or negate this signal until its transfer is complete. To avoid contention on the $\overline{BB}$ signal, masters should three-state this signal when it gets a logical 1 value. This situation implies that the connection of an external pull-up resistor is needed to ensure that a master acquiring the bus recognizes that the $\overline{BB}$ signal is negated, regardless of how many cycles have passed since the previous master relinquished the bus. Refer to Figure 13-21 for more information.

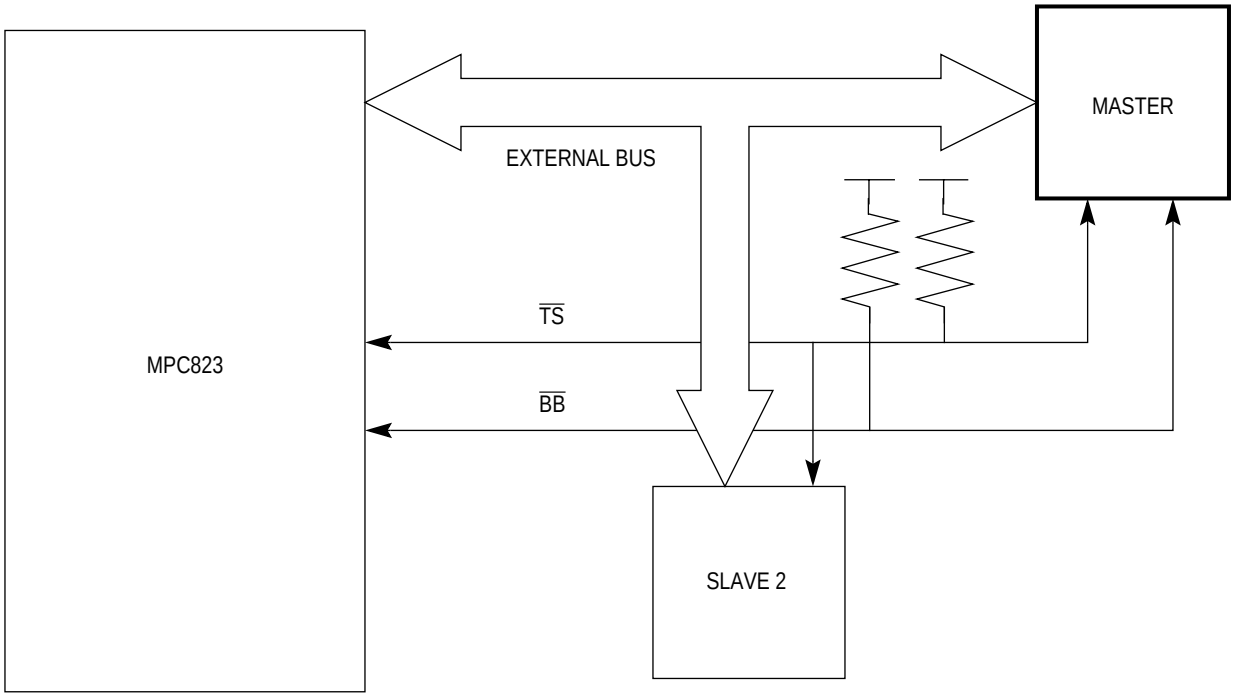

Figure 13-21. Basic Bus Busy Connection

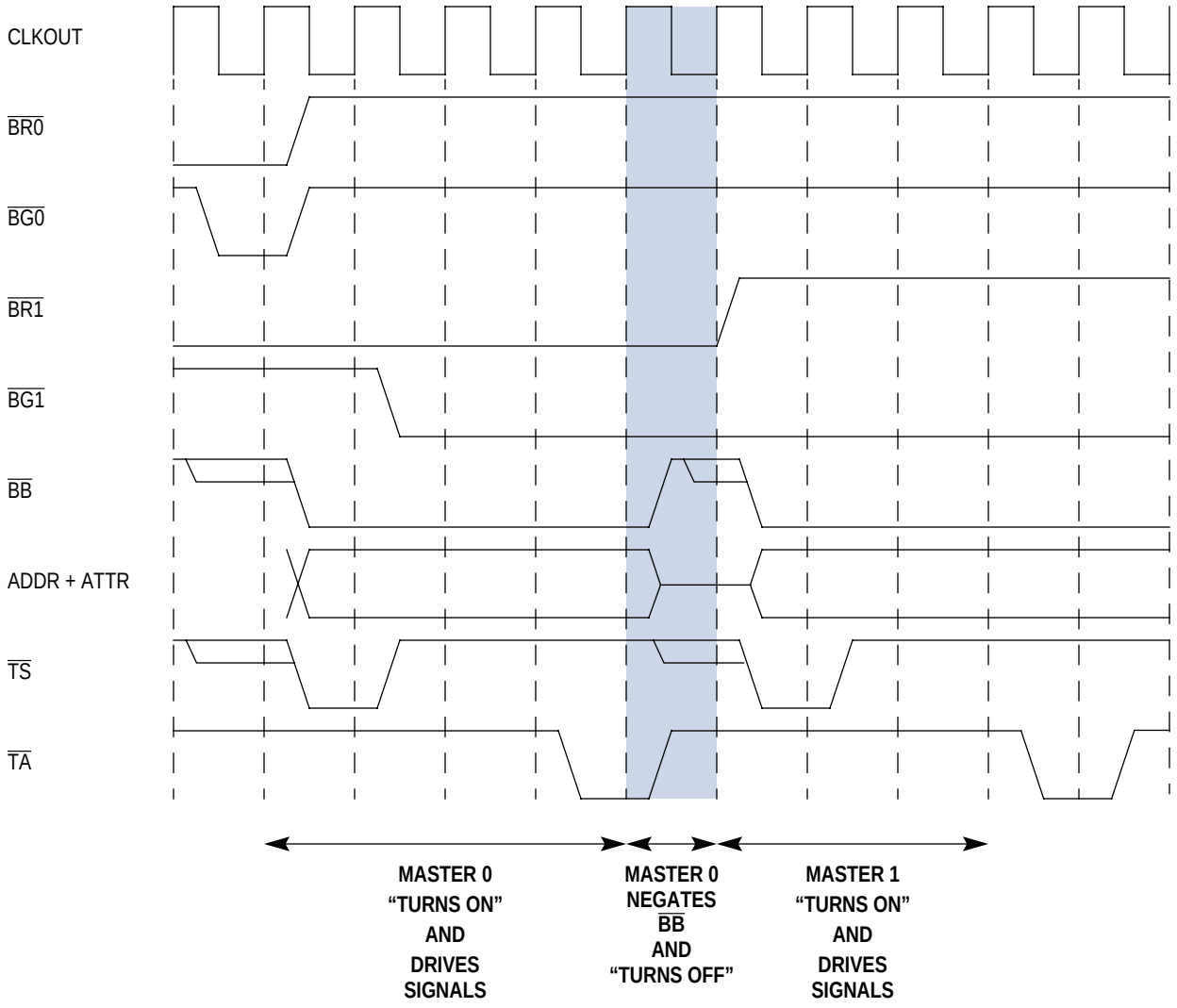

Figure 13-22. Bus Arbitration Timing Diagram

At system reset, the MPC823 can be configured to use the internal bus arbiter and it will be parked on the bus. The priority of the external device relative to the internal MPC823 bus masters is programmed in the SIU module configuration register, as shown in **Section 12.12.1.1 SIU Module Configuration Register**. If the external device requests the bus and the MPC823 does not need it or the external device has priority over the current internal bus master, the MPC823 grants the bus to the external device. Figure 13-23 illustrates the internal finite state machine that implements the arbiter protocol.

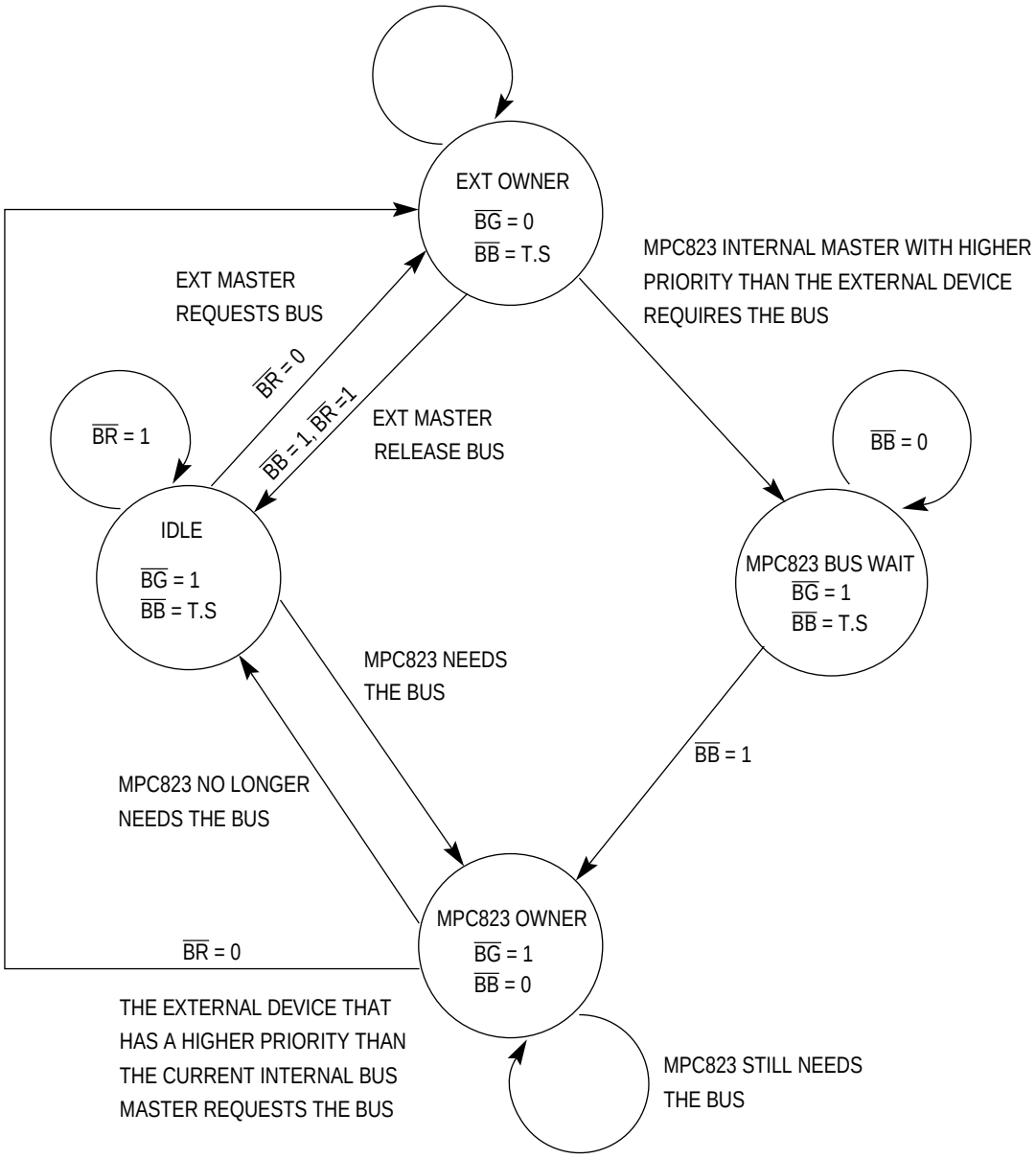

Figure 13-23. Internal Bus Arbitration State Machine

13.4.7 Address Transfer Phase-Related Signals

13.4.7.1 TRANSFER START SIGNAL. The $\overline{TS}$ signal indicates the beginning of a cycle initiated by the bus master. This signal should be asserted by a master only after ownership of the bus is granted by the arbitration protocol. This signal is only asserted for the first clock cycle of the transaction and is negated in the successive clock cycles. The master should three-state this signal when it relinquishes the bus to avoid contention between two or more masters in this signal. This configuration requires an external pull-up resistor to be connected to the $\overline{TS}$ signal. This will prevent a slave from responding to a bogus $\overline{TS}$ assertion. Refer back to Figure 13-21 for more information.

13.4.7.2 ADDRESS BUS. The 26-bit address bus consists of address bits 0–25 and Bit 6 is most-significant. The bus is byte addressable, so each address can address one or more bytes. The address and its attributes are driven on the bus with the $\overline{TS}$ signal and stay valid until the bus master received a $\overline{TA}$ signal from the slave. To distinguish the individual byte, the slave device must observe the TSIZx signals.

Note: Although this is a 32-bit machine, only 26 of the bits are visible outside the chip.

13.4.7.3 TRANSFER ATTRIBUTES. The transfer attributes consist of the RD/$\overline{WR}$, $\overline{BURST}$, TSIZx, ATx, $\overline{STS}$, and $\overline{BDIP}$ signals. These signals, except for the $\overline{BDIP}$, are available at the same time as the address bus.

13.4.7.3.1 Read/Write Signal. When the RD/$\overline{WR}$ signal is high it indicates a read access and when it is low it indicates a write access.

13.4.7.3.2 Burst Signal. The $\overline{BURST}$ signal and the address are driven by the bus master at the beginning of the bus cycle to indicate that the transfer is a burst transfer. The burst size is always 16 bytes. With a 32-bit port size, the burst includes 4 beats. When its port size is 16 bits and controlled by the internal memory controller, the burst includes 8 beats. When its port size is 8 bits and controlled by the internal memory controller, the burst includes 16 beats. The MPC823 bus supports critical data word first for burst. The order of the wraparound goes back to the critical word. For example, assuming data 2 is the critical word:

- Case burst of four beats:

  data 2 ➟ data 3 ➟ data 0 ➟ data 1

- Case burst of eight beats:

  data 2 ➟ data 3 ➟ data 4 ➟ ..... ➟ data 7 ➟ data 0 ➟ data 1

13.4.7.3.3 Transfer Size Signal. The TSIZx signals indicate the size of the requested data transfer and they can be used with the BURST and A[30:31] signals to determine which byte lanes of the data bus are involved in the transfer. For nonburst transfers, the TSIZx signals specify the number of bytes starting from the byte location addressed by the A[30:31] signals. In burst transfers, the value of the TSIZx signal is always 00.

Table 13-4. BURST/TSIZE Encoding

BURST	TSIZx	TRANSFER SIZE
1	01	Byte
1	10	Half-Word
1	11	x
1	00	Word
0	00	Burst (16 bytes)

13.4.7.3.4 Address Space Attributes. The address space attributes consist of the address type (AT[0:3]), PTR, and RSV signals, which are all outputs that indicate one of 16 “address types” to which the address applies. These types are designated as either a normal/alternate master cycle, problem/privilege (user or supervisor), and instruction or data types. The address space signals are valid at the rising edge of the clock in which the STS signal is asserted.

Address space signals reflect the current status of the master originating the access, not necessarily the status in which the original access to this location has occurred. An example of this situation is when a copyback of a dirty line in the data cache occurs after the privilege state of the processor has been changed since the last access to the same line. Functional usage of the ATx, PTR, and RSV signals is for the reservation protocol described in **Section 13.4.10 Storage Reservation Protocol**. Table 13-5 provides the space definition encoded by the STS, TS, ATx, PTR, and RSV signals.

Show cycles are accesses to the core’s internal bus devices. These accesses are driven externally for emulation, visibility, and debugging purposes. A show cycle can have one address phase and one data phase (or just an address phase for the instruction show cycles). The cycle can be a write or read access and the data for both the read and write accesses should be driven by the bus master. This is different than the normal bus read and write accesses. The address of the show cycle should be valid on the bus for one clock and the data of the show cycle should be valid on the bus for one clock. The data phase should not require a transfer acknowledge to terminate the bus-show cycle. In a burst show cycle only the first data beat will be shown externally.

13.4.7.3.5 Special Transfer Start Signal. The STS signal is driven by the MPC823 when it owns the external bus. It indicates the start of a transaction on the external bus or an internal transaction in show cycle mode.

Table 13-5. Address Space Definitions

$\overline{STS}$	$\overline{TS}$	AT[0]	AT[1]	AT[2]	AT[3]	$\overline{PTR}$	$\overline{RSV}$	DEFINITIONS
		CORE/CPM	PROBLEM STATE/ PRIVILEGE STATE	INSTRUCTION/ DATA	RESERVATION/ PROGRAM TRACE	PROGRAM TRACE	RESERVATION	
1	x	x	x	x	x	1	1	No Transfer or no first transaction of a transfer
0	x	x	x	x	x	x	x	Start of a transaction
x	0	0	0	0	0	0	1	Core, Normal Instruction, Program Trace, Privilege State
					1	1	1	Core, Normal Instruction, Privilege State
				1	0	1	0	Core, Reservation Data, Privilege State
					1	1	1	Core, Normal Data, Privilege State
			1	0	0	0	1	Core, Normal Instruction, Program Trace, Problem State
					1	1	1	Core, Normal Instruction, Problem State
				1	0	1	0	Core, Reservation Data, Problem State
					1	1	1	Core, Normal Data, Problem State
		1	CH0	CH1	CH2	1	1	No Core, Normal, (CH indicates channel number)

Table 13-5. Address Space Definitions (Continued)

$\overline{STS}$	$\overline{TS}$	AT[0]	AT[1]	AT[2]	AT[3]	$\overline{PTR}$	$\overline{RSV}$	DEFINITIONS
		CORE/CPM	PROBLEM STATE/ PRIVILEGE STATE	INSTRUCTION/ DATA	RESERVATION/ PROGRAM TRACE	PROGRAM TRACE	RESERVATION	
x	1	0	0	0	0	0	1	Core, Show Cycle Address Instruction, Program Trace, Privilege State
					1	1	1	Core, Show Cycle Address Instruction, Privilege State
				1	0	1	0	Core, Reservation Show Cycle Data, Privilege State
					1	1	1	Core, Show Cycle Data, Privilege State
			1	0	0	0	1	Core, Show Cycle Address Instruction, Program Trace, Problem State
					1	1	1	Core, Show Cycle Address Instruction, Problem State
				1	0	1	0	Core, Reservation Show Cycle Data, Problem State
					1	1	1	Core, Show Cycle Data, Problem State
		1	CH0	CH1	CH2	1	1	No Core, Show Cycle Data (CH indicates channel number)

13.4.7.3.6 Burst Data in Progress Signal. The $\overline{\text{BDIP}}$ signal is sent from the master to the slave to indicate that there is a data beat following the current data beat. The master uses this signal to give the slave advanced warning of the remaining data in the burst. By negating the $\overline{\text{BDIP}}$ signal, you can terminate a burst cycle early. Refer to **Section 13.4.2 Single Beat Transfers** and **Section 13.4.4 The Burst Mechanism** for more information.

13.4.8 Data Transfer Phase-Related Signals

13.4.8.1 DATA SIGNAL. The D[0:31] signals are driven by the MPC823 when it owns the external bus and has initiated a write transaction to a slave device. During a read transaction the D[0:31] signals are driven by the slave device. See Table 13-2 for byte lane assignments.

13.4.9 Termination Phase-Related Signals

13.4.9.1 TRANSFER ACKNOWLEDGE SIGNAL. The $\overline{\text{TA}}$ signal indicates the normal completion of a bus transfer. During burst cycles, the slave asserts this signal with every data beat returned or accepted. This signal should be pulled up to V_{DD} with a pull-up resistor.

13.4.9.2 BURST INHIBIT SIGNAL. The $\overline{\text{BI}}$ signal is sent from the slave to the master to indicate that the addressed device does not have burst capability. If this signal is asserted or equal to 0, the master must transfer in multiple cycles and increment the address for the slave to complete the burst transfer. For a system that does not use the burst mode at all, this signal can be permanently tied low. This signal should be pulled up to V_{DD} with a pull-up resistor.

13.4.9.3 TRANSFER ERROR ACKNOWLEDGE SIGNAL. The $\overline{\text{TEA}}$ signal terminates the bus cycle under bus error conditions. The current bus cycle should be aborted. This signal should override any other cycle termination signals, such as the $\overline{\text{TA}}$ signal. This signal should be pulled up to V_{DD} with a pull-up resistor.

13.4.9.4 PROTOCOL FOR TERMINATION SIGNALS. The transfer protocol was defined to avoid electrical contention on signals that can be driven by various sources. To do that, a slave should not drive signals associated with the data transfer until the address phase is completed and it recognizes the address as its own. The slave should disconnect from signals immediately after it has acknowledged the cycle and no later than the termination of the next address phase cycle. This indicates that the termination signals should be connected to power through a pull-up resistor to avoid a situation in which a master samples an undefined value in any of these signals when no real slave is addressed. See Figure 13-24 and Figure 13-25 for more information.

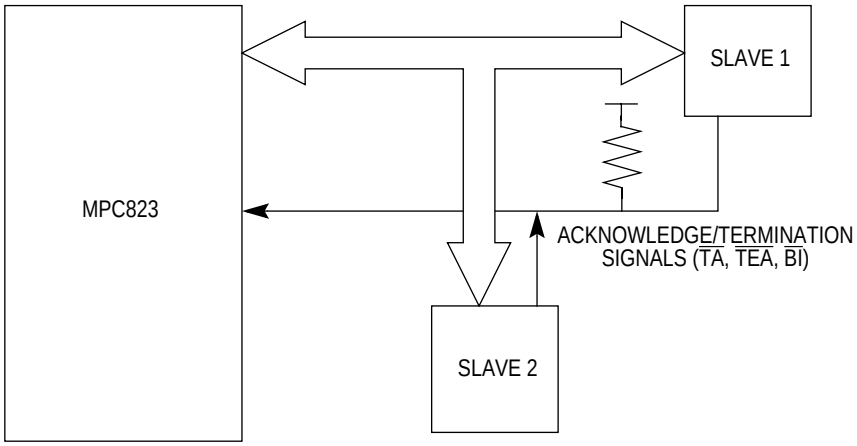

Figure 13-24. Termination Signals Protocol Basic Connection

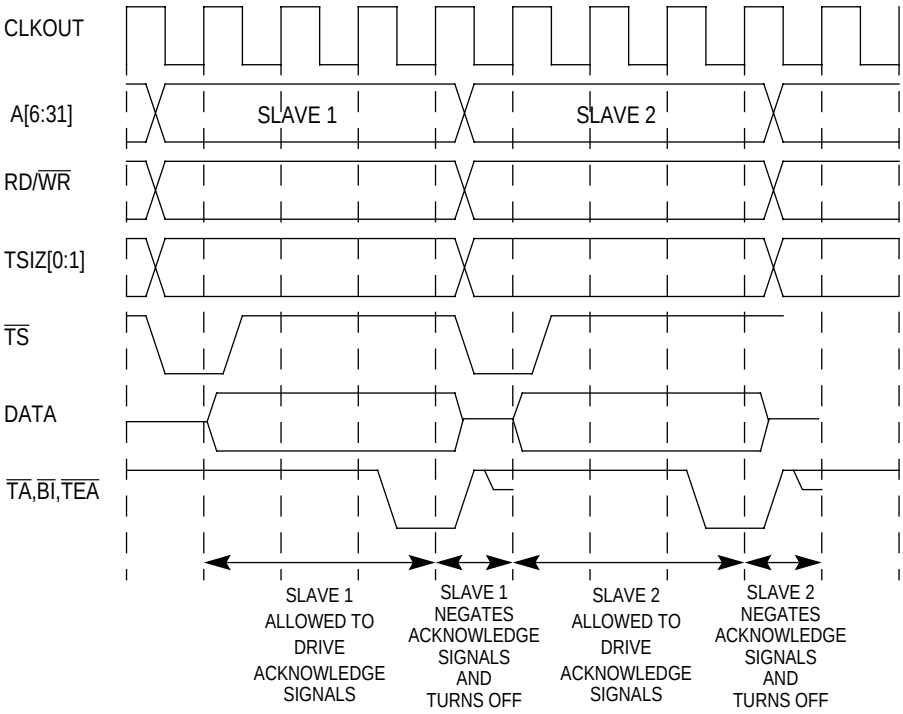

Figure 13-25. Termination Signals Protocol Timing Diagram

13.4.10 Storage Reservation Protocol

The MPC823 storage reservation protocol supports multilevel bus structure. For each local bus, storage reservation is handled by the local reservation logic. The protocol tries to optimize reservation cancellation so that a PowerPC processor is notified of storage reservation loss on a remote bus only when it has issued a **stwcx** cycle to that address. In other words, the reservation loss indication comes as part of the **stwcx** cycle. This method avoids the need to have fast storage reservation loss indication signals routed from every remote bus to every PowerPC master.

The storage reservation protocol makes the following assumptions:

- Each processor has, at most, one reservation “flag”
- **lwarx** sets the reservation “flag”
- **lwarx** by the same processor clears the reservation “flag” related to a previous **lwarx** instruction and again sets the reservation “flag”
- **stwcx** by the same processor clears the reservation “flag”
- A store by the same processor does not clear the reservation “flag”
- Some other processor (or other mechanism) store to the same address as an existing reservation clears the reservation “flag”
- If the storage reservation is lost it is guaranteed that **stwcx** will not modify the storage

The reservation protocol for a single-level (local) bus is illustrated in Figure 13-26. It assumes that external logic on the bus performs the following functions:

- Snoops accesses to all local bus slaves
- Holds one reservation for each local master capable of storage reservations
- Sets the reservation when that master issues a load and reserve request
- Clears the reservation when some other master issues a store to the reservation address

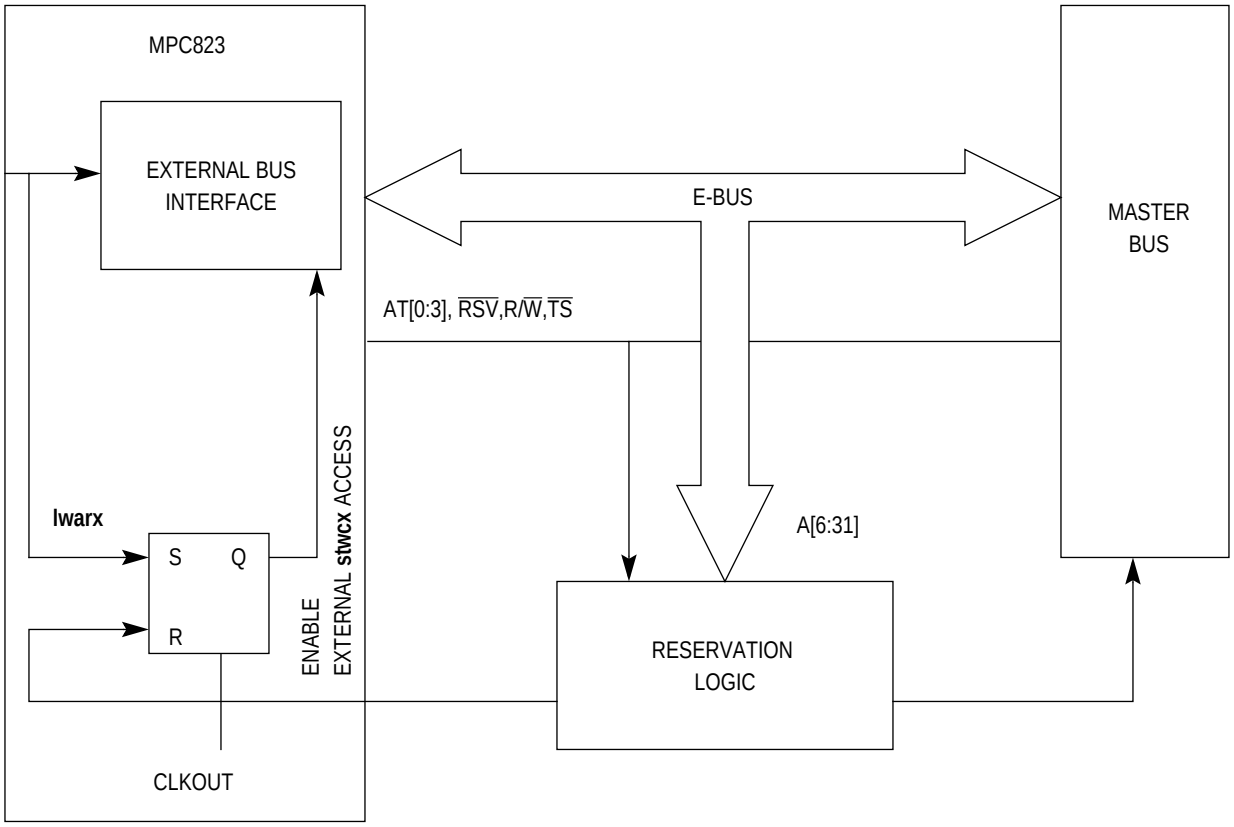

Figure 13-26. Reservation On Local Bus

The local bus interface block implements a reservation “flag” for the local bus master. The reservation “flag” is set by the local bus interface when a load with reservation is issued by the local bus master and the reservation address is located on the remote bus. The “flag” is reset when an alternative master on the remote bus accesses the same location in a write cycle. If the MPC823 begins a memory cycle to the previously reserved address (located in the remote bus) as a result of a **stwcx** instruction, one of the following conditions can occur:

- If the reservation “flag” is set, the local bus interface acknowledges the cycle in a normal way.
- If the reservation “flag” is reset, the local bus interface should assert $\overline{KR}$. However, the local bus interface should either not perform the remote bus write access or abort it if the remote bus supports aborted cycles. The failure of the **stwcx** instruction is reported to the core.

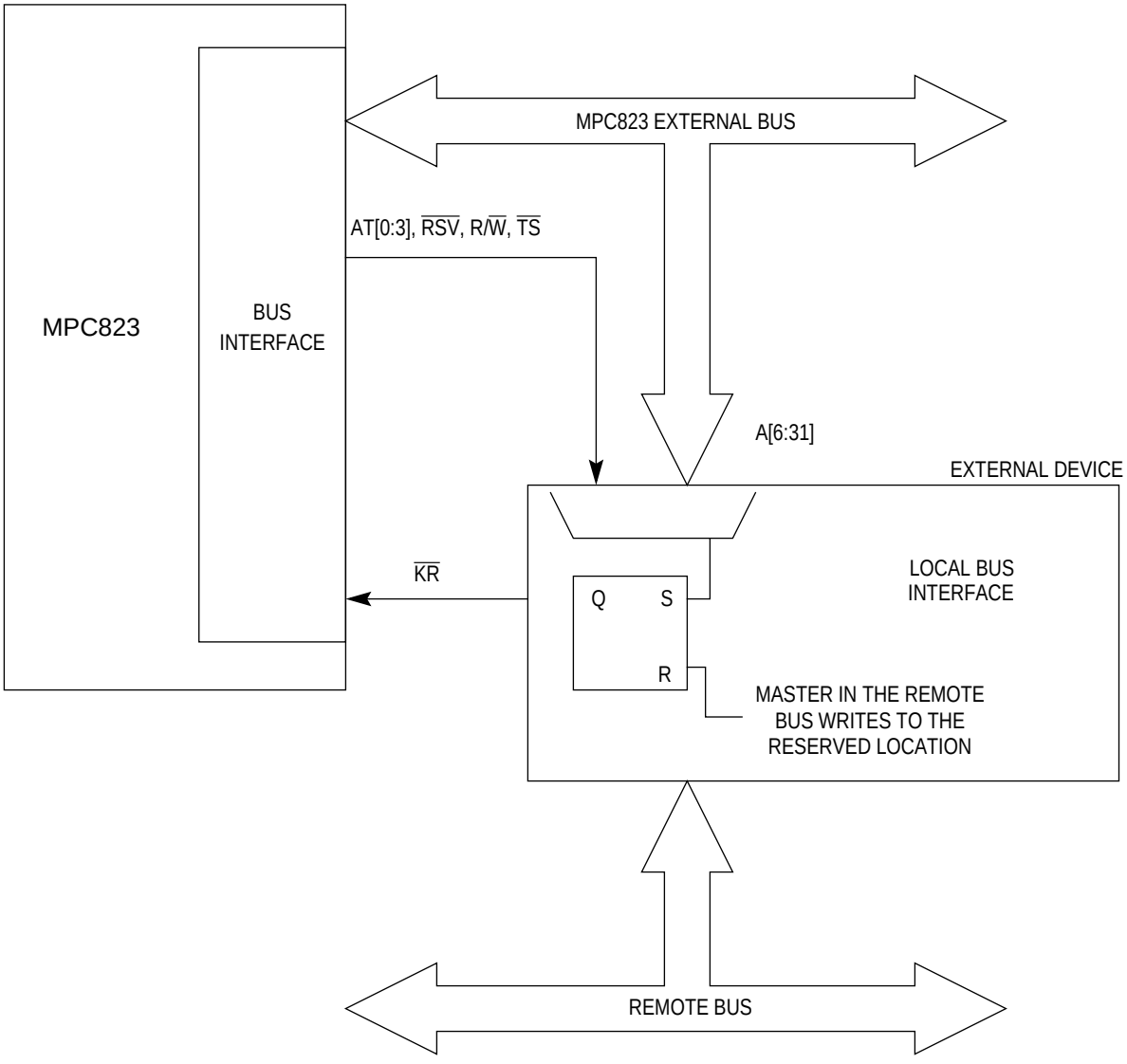

Figure 13-27. Reservation On Multilevel Bus Hierarchy

13.4.11 Exception Control Cycles

The MPC823 bus architecture requires the $\overline{TA}$ signal to be asserted from an external device to indicate that the bus cycle is complete. $\overline{TA}$ is not asserted when one of the following conditions occur:

- The external device does not respond
- Other application-dependent errors occur

The external circuitry can provide $\overline{TEA}$ when no device responds by asserting $\overline{TA}$ within an appropriate period of time after the MPC823 initiates the bus cycle. This allows the cycle to terminate and the processor to enter exception processing for the error condition.

To properly control termination of a bus cycle for a bus error, $\overline{TEA}$ must be asserted simultaneously or before $\overline{TA}$ is asserted. $\overline{TEA}$ should be negated before the second rising edge after it was sample-asserted to avoid detecting an error for the next initiated bus cycle. $\overline{TEA}$ is an open-drain pin that allows the wire-OR of any different error generation sources.

13.4.11.1 RETRY SIGNAL. When an external device asserts the $\overline{RETRY}$ signal during a bus cycle, the MPC823 enters a sequence in which it terminates the current transaction, relinquishes ownership of the bus, and retries the cycle using the same address, address attributes, and data. Figure 13-28 illustrates the behavior of the MPC823 when the $\overline{RETRY}$ signal is detected as a termination of a transfer. The figure illustrates that when the internal arbiter is enabled, the MPC823 negates the $\overline{BB}$ signal and asserts the $\overline{BG}$ signal in the clock cycle following retry detection. This allows any external master to gain bus ownership. In the next clock cycle, a normal arbitration procedure may occur. The figure also shows that the external master did not use the bus, so the MPC823 initiates a new transfer with the same address and attributes as before. In Figure 13-29 the same situation is illustrated to show that the MPC823 is working with an external arbiter. In the clock cycle after the CPU recognizes that the $\overline{RETRY}$ signal is asserted, the $\overline{BR}$ and $\overline{BB}$ signals are negated. One clock cycle later, the normal arbitration procedure may occur. This input signal requires a pull-up resistor.

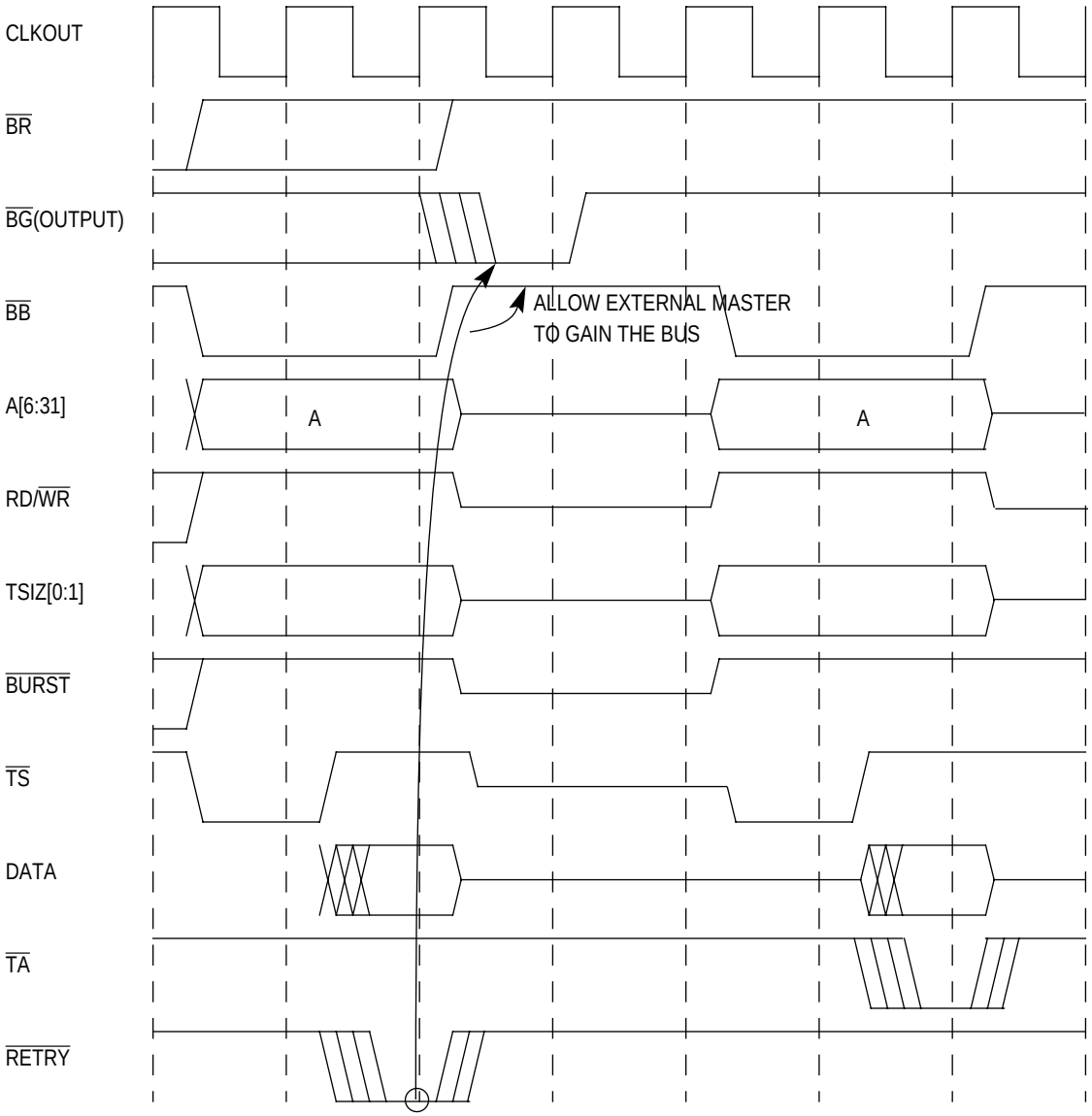

Figure 13-28. RETRY Transfer Timing–Internal Arbiter

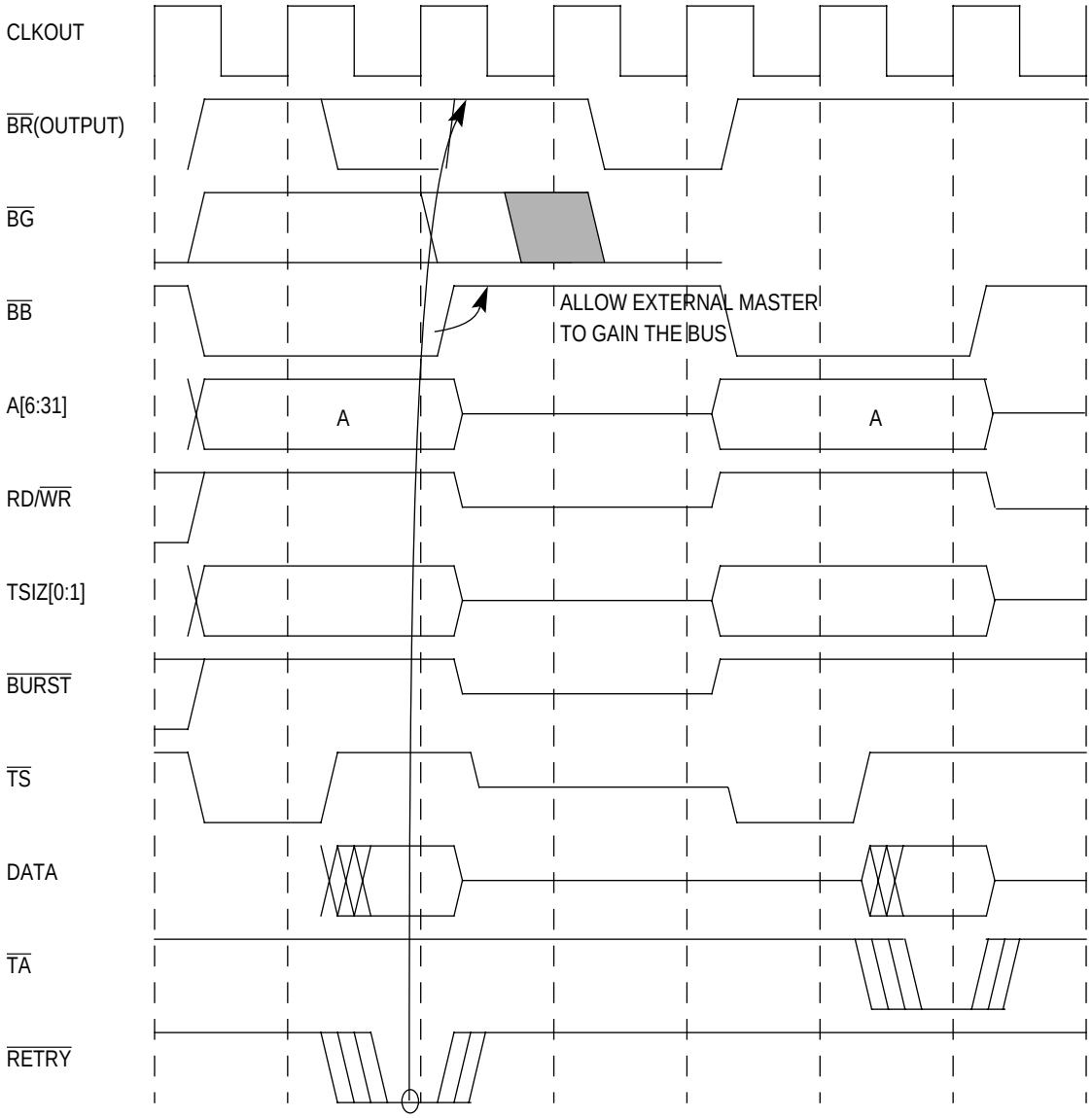

Figure 13-29. RETRY Transfer Timing–External Arbiter

When a burst access is initiated by the MPC823, the bus interface only recognizes the $\overline{\text{RETRY}}$ assertion as a retry termination if it detects it before the first data beat is acknowledged by the slave device. When the $\overline{\text{RETRY}}$ signal is asserted as a termination signal on the second or third data beat of the access, the MPC823 recognizes it as a transfer error acknowledgement.

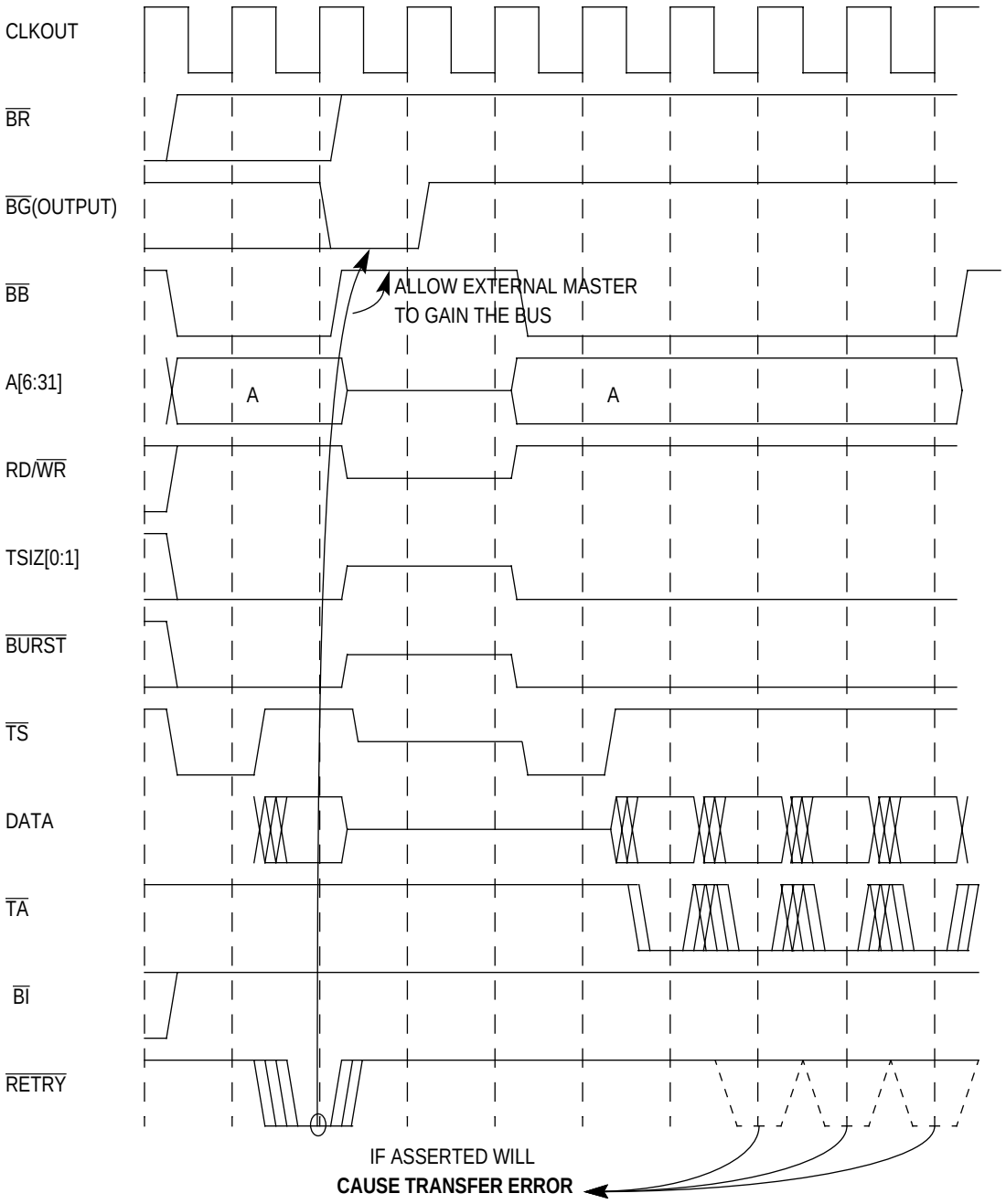

Figure 13-30. Retry On Burst Cycle

In reference to Figure 13-30, if the $\overline{BI}$ signal is asserted at the first beat of a burst, then the remaining beats of the 16-byte transfer retry are recognized as a transfer error acknowledge. Table 13-6 summarizes how the MPC823 recognizes the termination signals provided by the slave device that the initiated transfer addressed.

Table 13-6. Termination Signal Protocol

$\overline{TEA}$	$\overline{TA}$	$\overline{RETRY}/\overline{KR}$	ACTION
Asserted	X	X	Transfer Error Termination
Negated	Asserted	X	Normal Transfer Termination
Negated	Negated	Asserted	Retry Transfer Termination / Kill Reservation